

AD-A229 300

NRL Memorandum Report 6572

moves

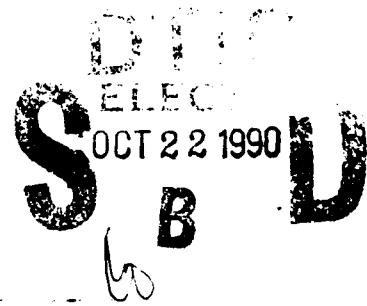
Estimation of The Time of Arrival
of Underwater Acoustic Signals
by Spline Functions
I: An Introduction

Charles K. Chui

Center for Approximation Theory
Texas A&M University
College Station, TX 77843

for

Underwater Sound Reference Detachment
Naval Research Laboratory
P.O. Box 568337
Orlando, FL 32856-8337



1 October 1990

Approved for public release; distribution unlimited

REPORT DOCUMENTATION PAGE				Form Approved OMB No 0704-0188
<p>Please report any burden for this collection of information, as estimated to average individual response, including the time for reviewing instructions, learning existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4102, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.</p>				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED		
	1 October 1990	Final	June 88 til Apr 90	
4. TITLE AND SUBTITLE		5. FUNDING NUMBERS		
Estimation of The Time-of-Arrival of Underwater Acoustic Signals by Spline Functions - I: An Introduction		PE #61153N Task #RR011-08-42		
6. AUTHOR(S)		WU Assn. #DN220-161 WU #59-0589-0-0		
Charles K. Chui				
Contracts: N62190-88-M-0409 & N62190-89-M-1536				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)		8. PERFORMING ORGANIZATION REPORT NUMBER		
Center for Approximation Theory Texas A&M University College Station, TX 77843				
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSORING / MONITORING AGENCY REPORT NUMBER		
Measurements Branch Underwater Sound Reference Detachment Naval Research Laboratory P.O. Box 568337, Orlando, FL 32856-8337		NRL Memorandum Report 6572		
11. SUPPLEMENTARY NOTES This work was supported by the Transducer Technology Project sponsored by the Office of Naval Technology (ONT 231). Part II of this work is reported in NRL Memorandum Report 6573 dated 1 Oct 90.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT		12b. DISTRIBUTION CODE		
Approved for public release; distribution unlimited.				
13. ABSTRACT (Maximum 200 words) The underwater acoustic signal is represented by a spline curve whose initial knot lies in the interval of measurement so that it may be used to determine the time-of-arrival of the acoustic signal. The first section is devoted to demonstrate a computational procedure of spline functions using the Bernstein representations, so that the coefficient matrices in the penalized least-squares problem can be determined much more efficiently. Since the coefficient matrices are usually singular when the discrete least-squares problem is considered, the second section discusses the singular value decomposition and Moore-Penroe inverses. The extremal problem that characterizes the time-of-arrival is discussed in Section 3, where existence, uniqueness, and characterization results are obtained. In particular, when linear splines are used, the extremal problem has a very elegant formulation that can be easily implemented in the computer. <i>Second: An Alternative Analysis (cont'd)</i>				
14. SUBJECT TERMS		15. NUMBER OF PAGES		
Underwater acoustic signals; time-of-arrival; spline representation; Moore-Penroe inverses; extremal problem; existence and uniqueness of solution; characterization and simplification; algorithms		39		
16. PRICE CODE				
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT	
Unclassified	Unclassified	Unclassified	U	

CONTENTS

INTRODUCTION	1
FITTING OF UNDERWATER ACOUSTIC SIGNALS BY SPLINE CURVES	1
Representation of Polynomials	2
Spline Functions	4
Computation of <i>B</i> -Splines	5
Spline Representation of Underwater Acoustic Signals	7
Computation of the Coefficient Matrices	11
THE MINIMUM-NORMED LEAST-SQUARES SOLUTION	16
Singular Value Decomposition and the Moore-Penrose Pseudoinverse	16
Characterization of the Moore-Penrose Pseudoinverse	20
Application to Least-Squares Estimation	22
ESTIMATION OF TIME-OF-ARRIVAL	23
Existence, Uniqueness, and Characterization	24
Estimation for the Continuous Setting	25
Linear Spline Estimate	27
Estimation with Splines of Arbitrary Degree	31
Estimation for the Discrete Setting	32
ACKNOWLEDGMENT	34
REFERENCES	34



Accession #	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification _____	
By _____	
Distribution/ _____	
Availability Codes _____	
Dist	Avail and/or Special
P-1	

(Blank Page)

ESTIMATION OF TIME-OF-ARRIVAL OF UNDERWATER ACOUSTIC SIGNALS BY SPLINE FUNCTIONS

INTRODUCTION

In the study of underwater acoustic signals, one of the most important problems is to determine the time-of-arrival (or signal onset) from the given data. For instance, the transducer and panel transfer function identification techniques all depend on an accurate estimate of this value. In this report, the underwater acoustic signal will be represented by a spline curve whose initial knot lies in the interior of the time interval so that the spline function is identically zero to the left of this knot and "takes off" to the right at this knot. Hence, the initial knot clearly defines the time-of-arrival of the acoustic signal. The objectives of this research are to study a mathematical model in the form of an extremal problem whose solution yields an accurate estimate of the time-of-arrival and to develop an algorithm for solving this extremal problem.

The first section will be devoted to the study of spline functions with special emphasis on the computational procedure of each polynomial piece exactly and determining the coefficient matrix in a modified (or "penalized") least-squares problem. Both the treatment and results here are different from those in the literature. When the discrete least-squares problem is solved, the coefficient matrix is usually singular. We will choose the solution whose least-squares measurement is minimized. This minimum solution is achieved by taking the so-called Moore-Penrose pseudoinverse of the coefficient matrix, a topic that will be discussed in the section entitled The Minimum-Normed Least Squares Solution. This presentation is intended to be complete so that both known and new results are included. The mathematical model that determines the initial knot, or equivalently the time-of-arrival of the acoustic signal, will be posed and studied in the section entitled Estimation of Time-of-Arrival. An algorithm to determine the solution of the corresponding extremal problem is described. A computer program with numerical examples will also be included.

FITTING OF UNDERWATER ACOUSTIC SIGNALS BY SPLINE CURVES

Underwater acoustic signals, noisy or not, can best be fitted by using spline curves. In addition to the usual benefits from spline representation such as efficiency in computation, flexibility in choosing the order of smoothness vs minimizing the possibility of oscillatory behaviour, the variation diminishing property, etc., the main reason in using spline curves to fit underwater acoustic signals is that the initial knot of the knot sequence of the approximating spline function clearly defines the time-of-arrival (or signal onset). This section is devoted to the study of spline functions. Although there is a vast literature on this subject, we will introduce a new approach which cannot be found in any published paper or book, except partially in Refs. 3 and 5, in order to facilitate our self-contained discussion and to improve the computational procedure to suit our purposes. In particular, since the initial knot is going to be an interior knot, the results in this section are somewhat different from those in the spline literature. This section is divided into five subsections. Since a spline function is a piecewise polynomial function, the first subsection contains a short discussion of representation of a polynomial by

“Bernstein coefficients.” One advantage is, of course, that a Bernstein polynomial defined by using the more general formulation here, is coordinate independent, so that it provides an ideal representation of any polynomial piece of a spline function between two consecutive knots. Spline functions are defined in the second subsection with emphasis on the ones with uniform knot sequences, and a computational method introduced in Ref. 5 using the Bernstein coefficients directly is included in the subsection entitled Computation of B -Splines. The last two subsections are important for the remaining material of this report. The new idea of representing an underwater acoustic signal by a spline curve with the first initial knot defining the time-of-arrival is introduced in the subsection entitled Spline Representation of Underwater Acoustic Signals. A modified (or penalized) least-squares procedure is used where the modification is achieved by introducing a parameter that governs the noise level. The final subsection includes a computational method of the coefficient matrices. This information is especially important for our computational task.

Representation of Polynomials

Let k be a non-negative integer and π_k denote the vector space of all real polynomials of degree at most k . That is, each $p \in \pi_k$ is of the form

$$p(x) = \sum_{j=0}^k a_j x^j \quad (1)$$

where a_0, \dots, a_k are real numbers. Of course, the collection $\{1, x, \dots, x^k\}$ of monomials provides a basis for π_k . However, for both theoretical and computational purposes, it is usually necessary to focus our attention on a certain interval $[a, b]$, and in doing so, the basis $\{1, x, \dots, x^k\}$ is no longer useful, since it does not reflect upon this interval. For this reason, we will use the so-called barycentric coordinate of the interval $[a, b]$ defined as follows. Let u and v be two linear polynomials defined by

$$u := u(x) = \frac{b-x}{b-a}, \quad v := v(x) = \frac{x-a}{b-a}. \quad (2)$$

Note that the pair (u, v) of “variables” identify the interval $[a, b]$ in the sense that u represents the initial end-point, a , relative to the interval $[a, b]$, and v the final end-point, b , relative to this interval, namely:

$$\begin{cases} u(a) = 1 \\ u(b) = 0 \end{cases} \quad \text{and} \quad \begin{cases} v(a) = 0 \\ v(b) = 1. \end{cases}$$

This pair of variables can be used in place of the variable x , since

$$x = ua + vb. \quad (3)$$

Of course, we must remind ourselves that u and v are related by the identity, $u + v = 1$ for all x . Now, it follows that the collection of $k+1$ polynomials

$$\varphi_{ij}^k(u, v) = \frac{k!}{i!j!} u^i v^j, \quad (4)$$

where $0 \leq i, j \leq k$ and $i + j = k$, is also a basis of π_k ; that is, every polynomial $p \in \pi_k$ can be uniquely represented as

$$p(u, v) = p(u(x), v(x)) = \sum_{i+j=k} a_{ij}^k \varphi_{ij}^k(u, v) \quad (5)$$

for all x , where the summation is taken over all i and j with the restrictions: $0 \leq i, j \leq k$ and $i + j = k$. In other words, the coefficients $\{a_{ij}^k\}$, $i + j = k$, called the *Bernstein coefficients* of $p(u, v)$ uniquely determine the polynomial. In Fig. 1, we display the Bernstein coefficients of three cubic polynomials on $[a, b]$, which are

$$\left(\frac{b-x}{b-a}\right)^3 + \left(\frac{x-a}{b-a}\right)^3,$$

$$3\left(\frac{b-x}{b-a}\right)\left(\frac{x-a}{b-a}\right)^2,$$

and

$$3\left(\frac{b-x}{b-a}\right)^2\left(\frac{x-a}{b-a}\right) - 3\left(\frac{b-x}{b-a}\right)\left(\frac{x-a}{b-a}\right)^2,$$

respectively. It is important to remark that not only the collection $\{a_{ij}^k\}$, $i + j = k$, uniquely determines $p(u, v)$, but it also gives a geometric description of the curve traced by the polynomial $p(u, v)$. Indeed, since $\varphi_{ij}^k \geq 0$ for all $x \in [a, b]$ (or equivalently, $0 \leq u, v \leq 1$) and $\sum \varphi_{ij}^k = 1$, $p(u, v)$ is a *convex combination* of its Bernstein coefficients a_{ij}^k , and hence, lies in the “convex hull” of the set

$$\left\{ \left(\frac{i}{k}a + \frac{j}{k}b \right), a_{ij}^k \right\}_{i+j=k} \quad (6)$$

This two-dimensional set is called the Bernstein net of $p(u, v)$. By joining this net by straight line segments, we note that the graph of $p(u, v)$ lies in the “convex hull” as shown in Fig. 2, where both the Bernstein nets and the polynomial curves of the examples given in Fig. 1 are shown.

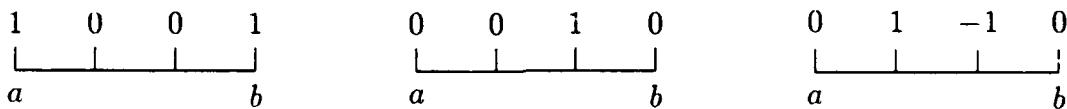


Fig. 1 - *B*-Coefficients of cubic polynomials

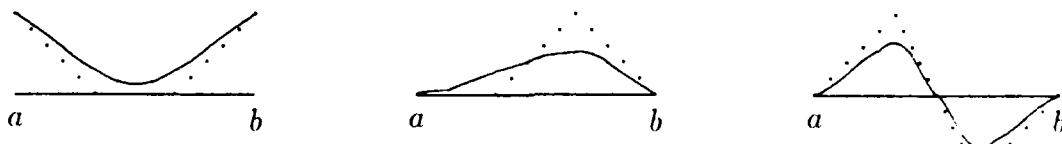


Fig. 2 - Graphs of cubic polynomials

Spline Functions

Let k be a non-negative integer and

$$\mathbf{t}: \dots < t_{-i} < \dots < t_i < \dots$$

be a bi-infinite sequence with $t_i \rightarrow \infty$ and $t_{-i} \rightarrow -\infty$ as $i \rightarrow \infty$. Suppose that $c = t_0$ and $d = t_n$ when n is a positive integer. Then a function $f \in C^{k-1}[c, d]$ (i.e., f has continuous derivatives up to order $k-1$ on the interval $[c, d]$) is called a *spline function of order* $(k+1)$ on $[c, d]$ with *knot sequence* \mathbf{t} , if the restriction of f on each interval $[t_i, t_{i+1}]$ is a polynomial of degree at most k , $i = \dots, -1, 0, 1, \dots$. We will denote the space of spline functions of order $(k+1)$ on $[c, d]$ with knot sequence \mathbf{t} by $S_{k,t}(c, d)$.

It is well known that a basis of the space $S_{k,t}(c, d)$ is given by the collection of so-called *B-splines* $B_{k,t,i}$, where $i = -k, \dots, n-1$, a totality of $n+k$ functions. Each *B-spline* $B_{k,t,i}$ is in $S_{k,t}(c, d)$ and vanishes identically outside the interval (t_i, t_{i+k+1}) . The interval $[t_i, t_{i+k+1}]$ is called the support of the *B-spline* $B_{k,t,i}$. It is also well known that the support $[t_i, t_{i+k+1}]$ is minimal in the sense that any function f in $S_{k,t}(c, d)$ whose support is a proper subset of $[t_i, t_{i+k+1}]$ must be the zero function [2]. In Fig. 3, we give the graphs of *B-splines* of order 1, 2, 3, and 4.

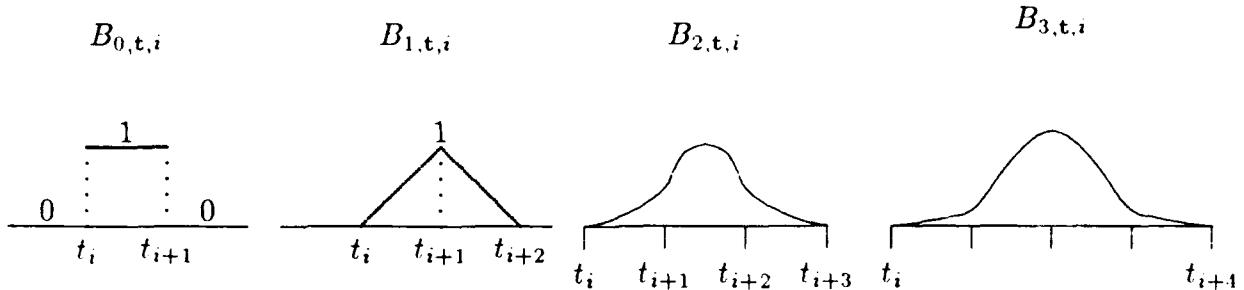


Fig. 3 - *B-Splines of order 1 - 4*

In this report, we are only concerned with spline spaces having uniform knot sequences \mathbf{t} ; that is, we only consider $t_{i+1} - t_i = t_i - t_{i-1}$ for all i . The construction of *B-splines* with uniform knot sequences is particularly easy. We start with the special case where $t_i = i$.

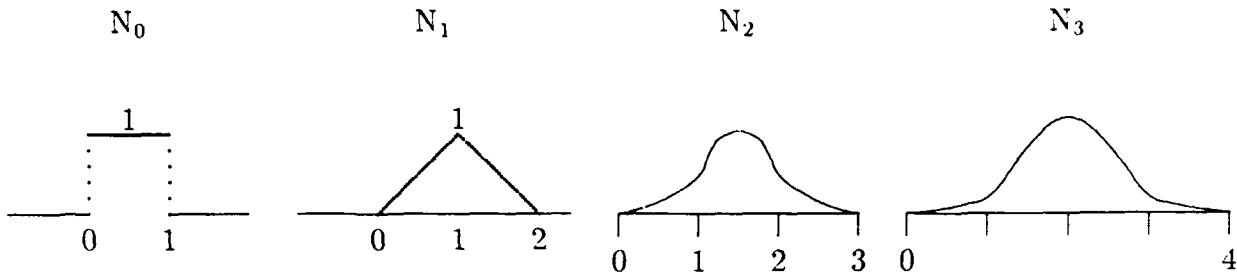
Let $N_0(x)$ be the characteristic function of the interval $(0,1)$; that is,

$$N_0(x) = \begin{cases} 1 & \text{if } 0 < x < 1 \\ 0 & \text{otherwise} \end{cases}. \quad (7a)$$

We define, inductively,

$$N_k(x) = \int_0^1 N_{k-1}(x-t)dt. \quad (7b)$$

$k = 1, 2, \dots$. The graphs of N_0, N_1, N_2 and N_3 are shown in Fig. 4.

Fig. 4 - *B*-Splines at integer knots

It is easy to show that $N_k \in C^{k-1}(-\infty, \infty)$, and since each integration increases the degree of the polynomial pieces by one, we see that N_k is in $S_{k,Z}$, where Z is the set of all integers. The *B*-splines in $S_{k,Z}$ are given by

$$B_{k,Z,i}(x) = N_k(x - i). \quad (8)$$

Now suppose that $t_{i+1} - t_i = h > 0$ for all i . Then the *B*-splines in $S_{k,t_h}(c, d)$, where the knot sequence is

$$\mathbf{t}_h: \dots < t_{-i} < \dots < t_i < \dots, \quad t_{i+1} - t_i = h,$$

with $t_0 = c$ and $t_n = d$, are clearly given by

$$B_{k,t_h,i}(x) = N_k\left(\frac{1}{h}(x - c) - i\right). \quad (9)$$

These are the *B*-splines that will be used in this report.

Computation of *B*-Splines

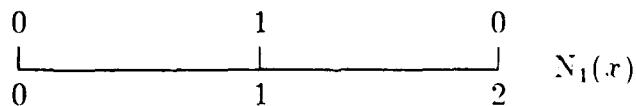
Let $t_{i+1} - t_i = h > 0$ for all i and

$$\mathbf{t}_h: \dots < t_{-i} < \dots < t_i < \dots$$

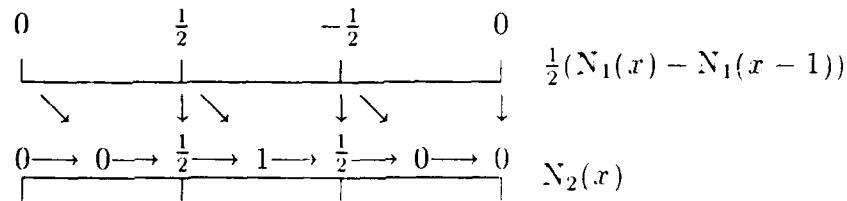
with $t_0 = c$ and $t_n = d$. In view of the formula in Eq. (9), to obtain the *B*-spline basis $\{B_{k,t_h,i}\}, i = -k, \dots, n - 1$, of the spline space $S_{k,t_h}(c, d)$, it is sufficient to determine $N_k(x)$. Although $N_k(x)$ can be computed by using the definition in Eqs. (7a) and (7b), it is more efficient to compute the Bernstein coefficients of each polynomial piece by following the method described in Ref. 3. Since $N_1(x)$ is piecewise linear with values

$$N_1(i) = \begin{cases} 1 & \text{if } i=1 \\ 0 & \text{if } i \neq 1 \end{cases} \quad (10)$$

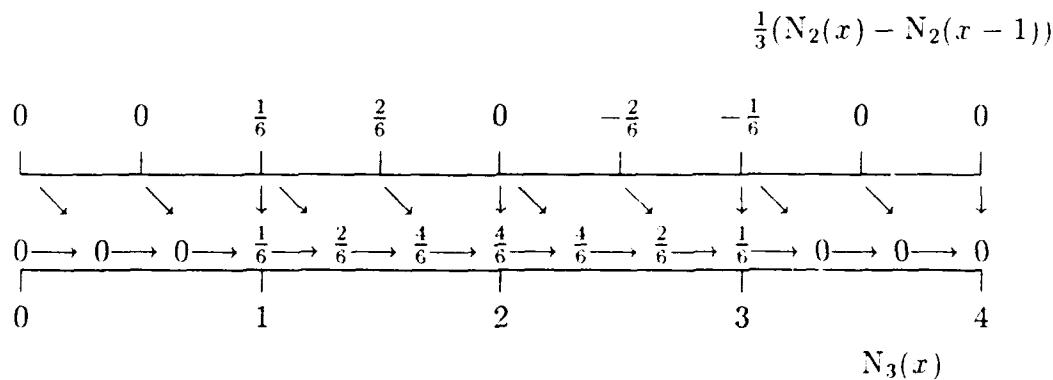
where $i \in Z$ (Fig. 4), its Bernstein coefficients are $\{0, 1\}$ and $\{1, 0\}$ on the intervals $[0, 1]$ and $[1, 2]$, respectively, as shown in Fig. 5.

Fig. 5 - *B*-Spline $N_1(x)$

To determine the Bernstein coefficients of $N_2(x)$, we need an intermediate step where the Bernstein coefficients of $(1/2)[N_1(x) - N_1(x-1)]$ are recorded. Now along the interval $[0,1]$, first input the zero initial condition. Then add this zero value to the first value of $(1/2)[N_1(x) - N_1(x-1)]$, namely zero, to get the second zero value. Next add this zero value to the second Bernstein coefficient of $(1/2)[N_1(x) - N_1(x-1)]$, namely $1/2$, to get the $1/2$ as the third Bernstein coefficient of $N_2(x)$ on $[0,1]$. For the interval $[1,2]$, we perform the same operation: add the initial value of $1/2$ to the first value, namely $1/2$, for $(1/2)[N_1(x) - N_1(x-1)]$, to get the second Bernstein coefficient of $N_2(x)$ on $[1,2]$ which is $(1/2) + (1/2) = 1$, and next, add this value 1 to the second value, namely $-1/2$ for $(1/2)[N_1(x) - N_1(x-1)]$, to get the third Bernstein coefficient of $N_2(x)$ on $[1,2]$ which is $1 - (1/2) = 1/2$. For the interval $[2,3]$, we repeat the same procedure. (See Fig. 6).

Fig. 6 - Construction of $N_2(x)$

To compute $N_3(x)$, we again need the intermediate step of $(1/3)[N_2(x) - N_2(x-1)]$. We then follow the same procedure as above in the computation of $N_2(x)$, but this time the support of $N_3(x)$ is $[0,4]$, one unit longer. (See Fig. 7).

Fig. 7 - Construction of $N_3(x)$

In Fig. 8, we show the computational procedure for the quartic and quintic splines $N_4(x)$ and $N_5(x)$, and for simplicity, we have omitted the arrows and write down the Bernstein nets for $N_3(x) - N_3(x-1)$ and $N_4(x) - N_4(x-1)$, instead of $(1/4)[N_3(x) - N_3(x-1)]$ and $(1/5)[N_4(x) - N_4(x-1)]$, respectively.

$$\begin{array}{ccccccccc}
 0 & 0 & 0 & \frac{1}{6} & \frac{2}{6} & \frac{4}{6} & \frac{4}{6} - \frac{1}{6} & \frac{4}{6} - \frac{2}{6} & \frac{2}{6} - \frac{4}{6} \\
 \hline
 & & & \frac{1}{6} & \frac{2}{6} & \frac{4}{6} & \frac{11}{6} & \frac{14}{6} & \frac{16}{6} \\
 & & & \frac{1}{24} & \frac{2}{24} & \frac{4}{24} & \frac{11}{24} & \frac{14}{24} & \frac{16}{24} \\
 \hline
 & & & & & & & & 0 & 0 & 0 & 0
 \end{array} \quad N_3(x) - N_3(x-1)$$

$$\begin{array}{ccccccccc}
 0 & 0 & 0 & 0 & \frac{1}{24} & \frac{2}{24} & \frac{4}{24} & \frac{8}{24} & \frac{11}{24} & \frac{14}{24} & \frac{16}{24} & \frac{14}{24} & \frac{11}{24} & \frac{8}{24} & \frac{4}{24} & \frac{2}{24} & \frac{1}{24} & 0 & 0 & 0 & 0
 \end{array} \quad N_4(x)$$

$$\begin{array}{ccccccccc}
 \cdots & \cdots & \cdots & \cdots & \frac{1}{120} & \frac{2}{120} & \frac{4}{120} & \frac{10}{120} & \frac{12}{120} & \frac{12}{120} & \frac{6}{120} & \cdots & \frac{6}{120} & \frac{-12}{120} & \frac{-12}{120} & \frac{-10}{120} & \frac{-6}{120} & \frac{-4}{120} & \frac{-2}{120} & \frac{-1}{120} & \cdots & \cdots & \cdots & \cdots
 \end{array} \quad N_4(x) - N_4(x-1)$$

$$\begin{array}{ccccccccc}
 \cdots & \cdots & \cdots & \cdots & \frac{1}{120} & \frac{2}{120} & \frac{4}{120} & \frac{10}{120} & \frac{12}{120} & \frac{12}{120} & \frac{6}{120} & \cdots & \frac{6}{120} & \frac{-12}{120} & \frac{-12}{120} & \frac{-10}{120} & \frac{-6}{120} & \frac{-4}{120} & \frac{-2}{120} & \frac{-1}{120} & \cdots & \cdots & \cdots & \cdots
 \end{array} \quad N_5(x)$$

Fig. 8 - Construction of $N_4(x)$ and $N_5(x)$

Spline Representation of Underwater Acoustic Signals

The spline model will be used in this report to represent underwater acoustic signals. Let the time interval in signal measurement be $[0, d]$. If $t_0 \geq 0$ is the time-of-arrival of an acoustic signal, then the spline curve that represents this signal must be identically zero for $t \leq t_0$ and "takes off" at $t = t_0$. Hence, the spline curve is given by the spline function

$$S_k(t) = \sum_{j=0}^{n-1} c_j B_{k,t_h,j}(t) \quad (11)$$

where the knot sequence is

$$t_h: \quad t_0 < t_1 < \cdots < t_{n-1} < \cdots < t_{n+k}$$

with $0 \leq t_0 = c$, $t_n = d$, and

$$t_j = t_{j-1} + h \quad , \quad h = \frac{d-c}{n}, \quad (12)$$

$j = 1, 2, \dots, n+k$. (See Fig. 9.)

Here, due to the nature of the spline series $S_k(t)$ in Eq. (11), this spline model assumes zero values of $S'_k(t_0), \dots, S_k^{k-1}(t_0)$ at the time-of-arrival t_0 . Hence, if the acoustic signal should have nonzero slope at the time-of-arrival, only the linear spline model ($k = 1$) can be used. In the forth coming report, we will allow stacked knots of spline functions of degree k at t_0 in order to make use of higher degree spline models even though the slope at the time-of-arrival may not be zero. In doing so, the procedure becomes adaptive in nature but involves more complicated computations.

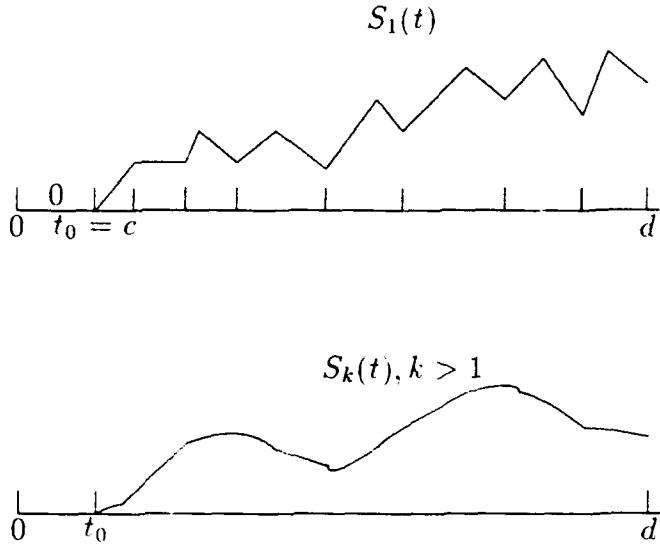


Fig. 9 - Spline model of signal

Let us assume that the measurement is taken at $\{\tau_i\}$, where $0 \leq \tau_1 < \dots < \tau_N \leq d$, and the signal measurement is

$$f(\tau_i) = f_i \quad , \quad i = 1, \dots, N.$$

In practice, we have $N \gg n$. The coefficients $\{c_j\}$ in the spline series $S_k(t)$ in Eq. (11) must be so chosen that the quantity

$$\|f - S_k\|^2 + \lambda \left(\sum_{j=0}^{n-1} c_j^2 \right) \quad (13)$$

is minimized, where $\| \cdot \|$ is a norm to be specified, and λ a non-negative parameter that is selected to adjust the "smoothness" of the acoustic spline curve $S_k(t)$. For noise-free signal $\{f_i\}$, λ should be chosen to be zero, but a positive λ value is required if the signal is contaminated with noise. In other words, the value of λ must be adjusted adaptively according to the noise level. A detailed study will be given in the next report [4].

Two practical norms $\| \cdot \|$ are the L^2 and $\ell^2(\mathbf{w})$ norms, defined respectively by:

$$\|g\|_{L^2} := \left(\int_0^d |g(t)|^2 dt \right)^{1/2} \quad (14)$$

and

$$\|g\|_{\ell^2(\mathbf{w})} := \left\{ \sum_{i=1}^N (g(\tau_i))^2 w_i \right\}^{1/2}, \quad (15)$$

where $\mathbf{w} = \{w_i\}$ is a given sequence of positive numbers, called weights. If the signal is noise-free, or if the variance of the noise process is time invariant, we may take $w_i = 1$.

On the other hand, for noisy data with time-varying variance, $\{w_i\}$ should be chosen according to the standard deviation of the noise. When the L^2 norm is used, the signal function $f(t)$ must be measured for all values of $t \in [0, d]$. This is usually not feasible, and one way to get around it is to define $f(t)$ as a piecewise linear function such that $\hat{f}(\tau_i) = f_i$, $i = 1, \dots, N$, and that the restriction of $\hat{f}(t)$ to each interval $[\tau_i, \tau_{i+1}]$ is a linear function, $1 \leq i \leq N$.

Let

$$F(\mathbf{c}) := F_h(\mathbf{c}) = \int_0^d |\hat{f}(t) - \sum_{j=0}^{n-1} c_j B_{k, t_h, j}(t)|^2 dt + \lambda \left(\sum_{j=0}^{n-1} c_j^2 \right). \quad (16)$$

For the time being let us assume that h is fixed (which is equivalent to saying that the time-of-arrival is assumed to be known). Here,

$$\mathbf{c} = \begin{bmatrix} c_0 \\ \vdots \\ c_{n-1} \end{bmatrix}. \quad (17)$$

To determine

$$\hat{S}_k(t) = \sum_{j=0}^{n-1} \hat{c}_j B_{k, t_h, j}(t) \quad (18)$$

which minimizes the quantity in Eq. (13), or equivalently $F(\mathbf{c})$ in Eq. (16), we simply determine the following “*normal equations*” by differentiating $F(\mathbf{c})$ with respect to \mathbf{c} :

$$\sum_{j=0}^{n-1} \left(\int_0^d B_{k, t_h, j}(t) B_{k, t_h, i}(t) dt \right) \hat{c}_j + \lambda \hat{c}_i = \int_0^d \hat{f}(t) B_{k, t_h, i}(t) dt, \quad (19)$$

$i = 0, \dots, n-1$. Hence, by setting

$$A_{k, h} = \left[\int_0^d B_{k, t_h, j}(t) B_{k, t_h, i}(t) dt \right]_{0 \leq i, j \leq n-1} \quad (20)$$

and

$$\mathbf{f}_h^0 = \begin{bmatrix} \int_0^d \hat{f}(t) B_{k, t_h, 0}(t) dt \\ \vdots \\ \int_0^d \hat{f}(t) B_{k, t_h, n-1}(t) dt \end{bmatrix}, \quad (21)$$

the system of normal equations Eq. (19) can be written in the matrix form:

$$(A_{k, h} + \lambda I_n) \hat{\mathbf{c}} = \mathbf{f}_h^0. \quad (22)$$

where I_n is the n^{th} order identity matrix and

$$\hat{\mathbf{c}} = \begin{bmatrix} \hat{c}_0 \\ \vdots \\ \hat{c}_{n-1} \end{bmatrix}. \quad (23)$$

One advantage of this approach is that since $A_{k,h}$ is the Gramian matrix of the B -splines and $\lambda \geq 0$, the coefficient matrix $(A_{k,h} + \lambda I_n)$ is nonsingular, so that $\hat{\mathbf{c}}$, and hence the solution $\hat{S}_k(t)$, can be uniquely determined.

A more standard approach is to use the $\ell^2(\mathbf{w})$ norm, so that the discrete data $\{(\tau_i, f_i)\}, i = 1, \dots, N$, can be input directly without introducing \hat{f} . The main advantage of this approach is that noisy data can be treated much more easily especially in the case when $N \gg n$. Let

$$G(\mathbf{c}) := G_h(\mathbf{c}) = \sum_{i=1}^N \left| f_i - \sum_{j=0}^{n-1} c_j B_{k,\mathbf{t}_h,j}(\tau_i) \right|^2 w_i + \lambda \sum_{j=0}^{n-1} c_j^2. \quad (24)$$

Again, for the time being let us assume that h is fixed. Then, to determine

$$S_k^*(t) = \sum_{j=0}^{n-1} c_j^* B_{k,\mathbf{t}_h,j}(t) \quad (25)$$

which minimizes the quantity in Eq. (13), or equivalently $G(\mathbf{c})$ in Eq. (24), we follow the same procedure as above to derive the following normal equations:

$$\sum_{j=0}^{n-1} \left(\sum_{\ell=1}^N B_{k,\mathbf{t}_h,j}(\tau_\ell) B_{k,\mathbf{t}_h,i}(\tau_\ell) w_\ell \right) c_j^* + \lambda c_i^* = \sum_{\ell=1}^N f_\ell B_{k,\mathbf{t}_h,i}(\tau_\ell) w_\ell, \quad (26)$$

$i = 0, \dots, n-1$; or in matrix form:

$$[\tilde{A}_{k,h} + \lambda I_n] \mathbf{c}^* = \tilde{\mathbf{f}}_h, \quad (27)$$

where $\tilde{A}_{k,h}$ is the $n \times n$ matrix whose $(i,j)^{\text{th}}$ entry is given by

$$\sum_{\ell=1}^N B_{k,\mathbf{t}_h,i}(\tau_\ell) B_{k,\mathbf{t}_h,j}(\tau_\ell) w_\ell,$$

and the vectors \mathbf{c}^* and $\tilde{\mathbf{f}}_h$ are

$$\mathbf{c}^* = \begin{bmatrix} c_0^* \\ \vdots \\ c_{n-1}^* \end{bmatrix} \quad (28)$$

and

$$\tilde{\mathbf{f}}_h = \begin{bmatrix} \sum_{\ell=1}^N f_\ell B_{k,\mathbf{t}_h,0}(\tau_\ell) w_\ell \\ \vdots \\ \sum_{\ell=1}^N f_\ell B_{k,\mathbf{t}_h,n-1}(\tau_\ell) w_\ell \end{bmatrix}. \quad (29)$$

respectively. The disadvantage of this approach is that the coefficient matrix $(\tilde{A}_{k,h} + \lambda I_n)$ is frequently singular so that the normal equation Eq. (27) might not have a unique solution. The standard approach to determine a solution \mathbf{c}^* of Eq. (27) is to consider another least-squares problem:

$$\min_{\mathbf{c}} \|(\tilde{A}_{k,h} + \lambda I_n)\mathbf{c} - \tilde{\mathbf{f}}_h\|_{\ell^2}.$$

However, even this problem usually does not have a unique solution. Hence, we will consider another extremal problem, by choosing \mathbf{c}^* to be the solution of the above least-squares problem with the minimum $\|\cdot\|_{\ell^2}$ value. It turns out that this "minimal solution" is now unique and can be determined by finding the so-called Moore-Penrose pseudoinverse of $(\tilde{A}_{k,h} + \lambda I_n)$. This topic will be discussed in the section entitled The Minimum-Normed Least Squares Solution.

Computation of the Coefficient Matrices

To solve the normal equations, Eqs. (22) and (27), it is necessary to determine the corresponding coefficient matrices. Hence, the quantities

$$b_{ij}^k = \int_0^d B_{k,t_h,i}(t)B_{k,t_h,j}(t)dt \quad (30)$$

and

$$\tilde{b}_{ij}^k = \sum_{\ell=1}^N B_{k,t_h,i}(\tau_\ell)B_{k,t_h,j}(\tau_\ell)w_\ell \quad (31)$$

must be computed. In this report, to produce a computational efficient algorithm, we have chosen the knot sequence \mathbf{t} to satisfy Eq. (12), so that the material presented in the subsection entitled Computation of B -Splines can be applied. In particular, by Eq. (26) we have

$$\begin{aligned} B_{k,t_h,i}(t) &= N_k\left(\frac{1}{h}(t - c) - i\right) \\ &= N_k\left(n - i - \frac{1}{h}(d - t)\right), \end{aligned} \quad (32)$$

so that

$$b_{ij}^k = h \int_0^{n-i} N_k(t)N_k(t+i-j)dt; \quad i, j = 0, \dots, n-1, \quad (33)$$

where we have used the property that $N_k(t) = 0$ for $t \leq 0$. Now the restriction of $N_k(t)$ on each interval $[r, r+1]$, where $r = 0, 1, \dots$, is a polynomial of degree k whose Bernstein coefficients can be computed by using the procedure outlined in the same subsection. Hence, to determine b_{ij}^k , we are led to compute

$$\int_r^{r+1} P_k Q_k,$$

where

$$P_k(u, v) = \sum_{\ell+m=k} c_{\ell m}^k \varphi_{\ell m}^k(u, v) \quad (34)$$

and

$$Q_k(u, v) = \sum_{p+q=k} d_{pq}^k \varphi_{pq}^k(u, v) \quad (35)$$

are two (Bernstein) polynomials of degree $\leq k$ on $[r, r+1]$. The computation is straightforward, since

$$\begin{aligned} \int_r^{r+1} \varphi_{\ell m}^k \varphi_{pq}^k &= \frac{k!}{\ell!m!} \frac{k!}{p!q!} \int_0^1 t^{(\ell+p)} (1-t)^{m+q} dt \\ &= \frac{k!}{\ell!m!} \frac{k!}{p!q!} \frac{\Gamma(\ell+p+1)\Gamma(m+q+1)}{\Gamma(\ell+p+m+q+2)} \\ &= \frac{k!}{\ell!m!} \frac{k!}{p!q!} \frac{(\ell+p)!(m+q)!}{(\ell+m+p+q+1)!} \\ &= \frac{(k!)^2}{(2k+1)!} \cdot \frac{(\ell+p)!(m+q)!}{\ell!m!p!q!}. \end{aligned} \quad (36)$$

Therefore, we have

$$\int_r^{r+1} P_k Q_k = \frac{(k!)^2}{(2k+1)!} \sum_{\ell+m=k} \sum_{p+q=k} \frac{(\ell+p)!(m+q)!}{\ell!m!p!q!} c_{\ell m}^k d_{pq}^k, \quad (37)$$

where $c_{\ell m}^k$ and d_{pq}^k are the Bernstein coefficients of P_k and Q_k respectively.

For example, if a linear spline curve is used, then by using the Bernstein coefficients of $N_1(x)$ in Fig. 5 and formulas in Eqs. (33) and (37), we have

$$b_{ij}^1 = \begin{cases} \frac{1}{3}h & \text{for } i=j=n-1 \\ \frac{2}{3}h & \text{for } i=j, i=0, \dots, n-2 \\ \frac{1}{6}h & \text{for } |i-j|=1 \\ 0 & \text{otherwise.} \end{cases} \quad (38)$$

If a cubic spline curve is used to fit the underwater acoustic signal, then the Bernstein coefficients of $N_3(x)$ in Fig. 7 and formulas in Eqs. (33) and (37) can be used to compute b_{ij}^3 , $0 \leq i, j \leq n$, as follows:

(a) For $i=j$ and $i=0, \dots, n-4$, we have

$$b_u^3 = 2 \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell c_p - d_\ell d_p] h$$

where $(c_0, \dots, c_3) = (0, 0, 0, 1/6)$ and $(d_0, \dots, d_3) = (1/6, 2/6, 4/6, 4/6)$, or

$$\begin{aligned}
 b_{ii}^3 &= \frac{2}{6^2} \frac{(3!)^2}{(7!)} \left\{ \frac{6!}{(3!)^2} + \left[\frac{6!}{(3!)^2} + \frac{2!4!}{(2!)^2} 4 + \frac{4!2!}{(2!)^2} 16 + \frac{6!}{(3!)^2} 16 \right. \right. \\
 &\quad + 2 \frac{5!}{2!3!} 2 + 2 \frac{2!4!}{3!2!} 4 + 2 \frac{(3!)^2}{(3!)^2} 4 \\
 &\quad \left. \left. + 2 \frac{(3!)^2}{(2!)^2} 8 + 2 \frac{4!2!}{2!3!} 8 + 2 \frac{5!}{2!3!} 16 \right] \right\} h = \frac{2416}{7!} h.
 \end{aligned}$$

(b) For $i = j = n - 3$, we have

$$\begin{aligned}
 b_{n-3,n-3}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell c_p + 2d_\ell d_p] h \\
 &= \frac{2396}{7!} h.
 \end{aligned}$$

(c) For $i = j = n - 2$, we have

$$\begin{aligned}
 b_{n-2,n-2}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell c_p + d_\ell d_p] h \\
 &= \frac{1208}{7!} h.
 \end{aligned}$$

(d) For $i = j = n - 1$, we have

$$\begin{aligned}
 b_{n-2,n-2}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell c_p] h \\
 &= \frac{20}{7!} h.
 \end{aligned}$$

(e) For $i = 0, \dots, n - 4$ and $j = i + 1$, we have

$$b_{ij}^3 = \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [2c_\ell d_p + d_\ell \bar{d}_p] h.$$

where $(\bar{d}_0, \bar{d}_1, \bar{d}_2, \bar{d}_3) = (4/6, 4/6, 2/6, 1/6)$; or

$$\begin{aligned}
 b_{ij}^3 &= \frac{1}{6^2} \frac{(3!)^2}{7!} \left\{ 2 \left[\frac{(3!)^2}{(3!)^2} + \frac{4!2!}{3!2!} 2 + \frac{5!}{3!2!} 4 + \frac{6!}{(3!)^2} 4 \right] \right. \\
 &\quad + \left[\frac{6!}{(3!)^2} 4 + \frac{2!4!}{(2!)^2} 8 + \frac{4!2!}{(2!)^2} 8 + \frac{6!}{(3!)^2} 4 \right. \\
 &\quad + \frac{5!}{3!2!} 12 + \frac{2!4!}{3!2!} 18 + \frac{(3!)^2}{(3!)^2} 17 \\
 &\quad \left. \left. + \frac{(3!)^2}{2!2!} 20 + \frac{4!2!}{2!3!} 18 + \frac{5!}{2!3!} 12 \right] \right\} h \\
 &= \frac{991}{7!} h.
 \end{aligned}$$

(f) For $i = n - 3$ and $j = n - 2$, we have

$$\begin{aligned} b_{n-3,n-2}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell d_p + d_\ell d_p] h \\ &= \frac{1062}{7!} h. \end{aligned}$$

(g) For $i = n - 2$ and $j = n - 1$, we have

$$\begin{aligned} b_{n-2,n-1}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell d_p] h \\ &= \frac{129}{7!} h. \end{aligned}$$

(h) For $i = 0, \dots, n-4$ and $j = i+2$, we have

$$\begin{aligned} b_{ij}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [2c_\ell d_p] h \\ &= \frac{1}{7!} 2 \left[\frac{(3!)^2}{(3!)^2} 4 + \frac{4!2!}{2!3!} 4 + \frac{5!}{2!3!} 2 + \frac{6!}{(3!)^2} \right] h \\ &= \frac{120}{7!} h. \end{aligned}$$

(i) For $i = n - 3$ and $j = n - 1$, we have

$$\begin{aligned} b_{n-3,n-1}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell d_p] h \\ &= \frac{60}{7!} h. \end{aligned}$$

(j) For $i = 0, \dots, n-4$ and $j = i+3$, we have

$$\begin{aligned} b_{ij}^3 &= \frac{(3!)^2}{7!} \sum_{\ell=0}^3 \sum_{p=0}^3 \frac{(\ell+p)!(6-\ell-p)!}{\ell!(3-\ell)!p!(3-p)!} [c_\ell c_\ell] h \\ &= \frac{(3!)^2}{7!} \frac{(3!)(3!)}{(3!)(3!)} \frac{1}{6^2} h = \frac{1}{7!} h, \end{aligned}$$

where $(c_0, \dots, c_3) = (1/6, 0, 0, 0)$.

(k) For $|i-j| \geq 4$, $b_{ij}^3 = 0$; and $b_{ji}^3 = b_{ij}^3$ by the definition in Eq. (30).

Summarizing the above results, we may write down the matrices $A_{k,h}$ for $k = 1$ and 3 as follows:

$$A_{1,h} = \frac{h}{6} \begin{bmatrix} 4 & 1 & & & \\ 1 & 4 & \ddots & & \\ & \ddots & \ddots & \ddots & \bigcirc \\ & & \ddots & \ddots & \ddots \\ \bigcirc & & & & \ddots \\ & & & & & 4 & 1 \\ & & & & & 1 & 2 \end{bmatrix}, \quad (39)$$

and

$$A_{3,h} = \frac{h}{7!} \begin{bmatrix} C_{n-3} & D \\ D^T & E_3 \end{bmatrix} \quad (40)$$

where C_{n-3} is an $(n - 3) \times (n - 3)$ banded Toeplitz symmetric matrix, D an $(n - 3) \times 3$ matrix with transpose D^T , and E_3 a 3×3 matrix given by:

$$C_{n-3} = \begin{bmatrix} 2416 & 991 & 120 & 1 & & \\ 991 & 2416 & 991 & \cdot & \cdot & \cdot & \circ \\ 120 & 991 & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 120 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 991 \\ \circ & \cdot & \cdot & \cdot & \cdot & \cdot & \end{bmatrix} \quad (41)$$

$$D = \begin{bmatrix} & \textcircled{O} & \\ 0 & & \\ 1 & 0 & \\ 120 & 1 & 0 \\ 991 & 120 & 1 \end{bmatrix} \quad (42)$$

$$E_3 = \begin{bmatrix} 2396 & 1062 & 60 \\ 1062 & 1208 & 129 \\ 60 & 129 & 20 \end{bmatrix}. \quad (43)$$

To compute \tilde{b}_{ij}^k , we note that again $\tilde{b}_{ji}^k = \tilde{b}_{ij}^k$ for all $i, j = 0, \dots, n - 1$ and

$$\tilde{b}_{ij}^k = \sum_{\ell=1}^N N_k \left(n - i - \frac{1}{h}(d - \tau_\ell) \right) N_k \left(n - j - \frac{1}{h}(d - \tau_\ell) \right) \quad (44)$$

where again $N_k(x)$ can be computed by using the procedure outlined in the subsection entitled Computation of B -Splines. Note that $N_k(x)$ vanishes outside the interval $[0, k + 1]$, so that most of the terms in the summation in Eq. (44) are zero. In particular, we again give the Bernstein coefficients of the linear B -spline $N_1(x)$ and the cubic B -spline $N_3(x)$ in the following:

$$\begin{array}{ccccccccc} 0 & & 1 & & 0 & & & & \\ \hline 0 & & 1 & & 2 & & & & \\ & & & & & & & & \\ 0 & 0 & 0 & \frac{1}{6} & \frac{2}{6} & \frac{4}{6} & \frac{4}{6} & \frac{4}{6} & \frac{2}{6} & \frac{1}{6} & 0 & 0 & 0 \\ \hline 0 & & & & & & & & & & & & & 4 \end{array} \quad N_1(x) \quad N_2(x)$$

THE MINIMUM-NORMED LEAST SQUARES SOLUTION

When discrete data information is given, a standard norm to use for the quantity described in Eq. (13) to be minimized is the $\ell^2(\mathbf{w})$ norm. For $\mathbf{w} = \{w_i\}$ with $w_i = 1$, we simply write $l^2 = l^2(\mathbf{w})$. Hence, for any sequence $\mathbf{c} = \{c_k\}$, we have

$$\|\mathbf{c}\|_{\ell^2} := \left\{ \sum_{i=1}^n c_i^2 \right\}^{1/2}. \quad (45)$$

As shown in the subsection entitled Spline Representation of Underwater Acoustic Signals, the corresponding extremal problem then reduces to a linear system described by Eq. (27), where the coefficient matrix $(\tilde{A}_{k,h} + \lambda I_n)$ is frequently singular. Hence, the linear system, Eq. (27), does not have a unique solution and is even “numerically inconsistent”. To overcome this, the usual method is to minimize the ℓ^2 norm of the difference $(\tilde{A}_{k,h} + \lambda I_n)\mathbf{c} - \tilde{\mathbf{f}}_h$. For convenience, we will simplify the notation by setting

$$A = \tilde{A}_{k,h} + \lambda I_n \quad , \quad \mathbf{b} = \tilde{\mathbf{f}}_h. \quad (46)$$

Hence, we will consider the extremal problem:

$$\min_{\mathbf{c}} \|A\mathbf{c} - \mathbf{b}\|_{\ell^2}. \quad (47)$$

Of course, if the original system $A\mathbf{c} = \mathbf{b}$ is consistent, then the minimum value in Eq. (47) is zero, and a solution to the extremal problem in Eq. (47) also solves the linear system $A\mathbf{c} = \mathbf{b}$ as required. In any case, whether the linear system $A\mathbf{c} = \mathbf{b}$ is consistent or not, there is no guarantee of a unique solution to the problem in Eq. (47). For various reasons such as stability (when n is very large), the desirable solution to Eq. (47) is one whose ℓ^2 norm is also minimized. That is, we will consider the problem:

$$\min\{\|\hat{\mathbf{c}}\|_{\ell^2}: \|A\hat{\mathbf{c}} - \mathbf{b}\|_{\ell^2} = \min_{\mathbf{c}} \|A\mathbf{c} - \mathbf{b}\|_{\ell^2}\}. \quad (48)$$

We will then choose the solution of Eq. (48) to be the solution \mathbf{c}^* in Eq. (27).

In this section, we will see that

$$\mathbf{c}^* = A^+ \mathbf{b} \quad \text{or} \quad \mathbf{c}^* = (\tilde{A}_{k,h} + \lambda I_n)^+ \tilde{\mathbf{f}}_h, \quad (49)$$

where A^+ is the so-called Moore-Penrose pseudoinverse of A .

Singular Value Decomposition and Moore-Penrose Pseudoinverse

The singular value decomposition of an arbitrary matrix is studied in this section. We will take a somewhat unusual route in introducing this familiar concept so that the definition of the Moore-Penrose pseudoinverse becomes very natural. We will also prove that this definition is independent of the (non-unique) singular value decomposition.

Let A be an $m \times n$ matrix which may not even be a square matrix (although in our application, it is always square and symmetric). Suppose that $\text{rank } A = k$, where $k \leq \min(m, n)$. Then $A^T A$ and $A A^T$ are $n \times n$ and $m \times m$ (respectively) non-negative definite symmetric matrices of rank k and have the same eigenvalues

$$\sigma_1^2 \geq \cdots \geq \sigma_k^2 > 0 \geq \cdots \geq 0,$$

which we arrange in non-increasing order. Let

$$\{\mathbf{v}_1, \dots, \mathbf{v}_n\}$$

be the eigenvectors of $A^T A$ corresponding to the eigenvalues $\{\sigma_1^2, \dots, \sigma_k^2, 0, \dots, 0\}$, so chosen that they form an orthonormal set. In other words, the $n \times n$ matrix

$$V = [\mathbf{v}_1 \dots \mathbf{v}_n]$$

is unitary, and

$$A^T A V = \begin{bmatrix} \sigma_1^2 & & & & \\ & \ddots & & & \\ & & \sigma_k^2 & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix} V. \quad (50)$$

Next, let σ_i be the positive square root of σ_i^2 and set

$$\mathbf{u}_i = \frac{1}{\sigma_i} A \mathbf{v}_i, \quad i = 1, \dots, k. \quad (51)$$

Then we have

$$A^T \mathbf{u}_i = \frac{1}{\sigma_i} (A^T A \mathbf{v}_i) = \frac{1}{\sigma_i} \sigma_i^2 \mathbf{v}_i, \quad (52)$$

or equivalently,

$$\mathbf{v}_i = \frac{1}{\sigma_i} A^T \mathbf{u}_i, \quad i = 1, \dots, k. \quad (53)$$

In addition, we have

$$A A^T \mathbf{u}_i = \sigma_i A \mathbf{v}_i = \sigma_i^2 \mathbf{u}_i, \quad (54)$$

so that \mathbf{u}_i is an eigenvector of $A A^T$ corresponding to the eigenvalue σ_i^2 , where $i = 1, \dots, k$. Let $\mathbf{u}_{k+1}, \dots, \mathbf{u}_m$ be orthonormal eigenvectors corresponding to the zero eigenvalues of $A A^T$; that is,

$$A A^T \mathbf{u}_i = 0, \quad i = k + 1, \dots, m,$$

and $\mathbf{u}_i^T \mathbf{u}_j = \delta_{ij}$, $i, j = k + 1, \dots, m$. Here, δ_{ij} is the Kronecker delta:

$$\delta_{ij} = \begin{cases} 1 & \text{for } i=j \\ 0 & \text{for } i \neq j. \end{cases}$$

Now, since $A A^T$ is symmetric, eigenvectors belonging to distinct eigenvalues are orthogonal, so that

$$\mathbf{u}_i^T \mathbf{u}_j = 0 \quad \text{for } i = 1, \dots, k \quad \text{and} \quad j = k + 1, \dots, m.$$

Next, for $i, j = 1, \dots, k$, we also have

$$\begin{aligned}\mathbf{u}_i^T \mathbf{u}_j &= \left(\frac{1}{\sigma_j} A \mathbf{v}_i \right)^T \left(\frac{1}{\sigma_j} A \mathbf{v}_j \right) \\ &= \frac{1}{\sigma_i \sigma_j} \mathbf{v}_i^T (A^T A \mathbf{v}_j) = \frac{1}{\sigma_i \sigma_j} \mathbf{v}_i^T (\sigma_j^2 \mathbf{v}_j) \\ &= \delta_{ij}\end{aligned}$$

since $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ is an orthonormal set. Summarizing the above properties of $\{\mathbf{u}_1, \dots, \mathbf{u}_m\}$, we see that the $m \times m$ matrix

$$U = [\mathbf{u}_1 \dots \mathbf{u}_m] \quad (55)$$

is unitary and

$$AA^T U = \begin{bmatrix} \sigma_1^2 & & & & \\ & \ddots & & & \\ & & \sigma_k^2 & & \\ & & & 0 & \\ & & & & \ddots \\ & & \textcircled{O} & & \\ & & & & 0 \end{bmatrix} U. \quad (56)$$

We also have the following.

Lemma 1. The matrix A has the following “singular value decomposition”:

$$A = U\Sigma V^T \quad (57)$$

where

$$\Sigma = \begin{bmatrix} \sigma_1 & & \textcircled{O} & \vdots & \\ & \ddots & & & \\ \textcircled{O} & & \sigma_k & \vdots & \textcircled{O} \\ \dots & \dots & \dots & \dots & \dots \\ \textcircled{O} & & \vdots & & \textcircled{O} \end{bmatrix}_{m \times n}. \quad (58)$$

Proof. Since V is unitary, Eq. (57) is equivalent to $AV = U\Sigma$, or equivalently:

$$A\mathbf{v}_1 = \sigma_1 \mathbf{u}_1, \dots, A\mathbf{v}_k = \sigma_k \mathbf{u}_k \quad (59)$$

since $A\mathbf{v}_{k+1} = \dots = A\mathbf{v}_n = 0$ and

$$U\Sigma = [\sigma_1 \mathbf{u}_1 \dots \sigma_k \mathbf{u}_k \ 0 \dots 0].$$

This completes the proof of the lemma since Eq. (59) is the same as Eq. (51).

In view of the result in Eq. (57) and the fact that $U^T = U^{-1}$ and $V^T = V^{-1}$, we are now ready to give a definition of the “inverse” of the matrix A (which is not necessarily square and not necessarily of full rank).

Definition. The Moore-Penrose pseudoinverse of an $m \times n$ matrix A is given by

$$A^+ = V\Sigma^+U^T \quad (60)$$

where

$$\Sigma^+ = \begin{bmatrix} \sigma_1^{-1} & & \circ & \vdots \\ & \ddots & & \vdots \\ \circ & & \sigma_k^{-1} & \vdots \\ \cdots & \cdots & \cdots & \cdots \\ & \circ & & \vdots & \circ \end{bmatrix}_{n \times m}. \quad (61)$$

Since the unitary matrices U and V are not unique, we must prove that the definition in Eq. (60) is independent of the choices of U and V . We have the following.

Theorem 1. Let $A = U\Sigma V^T = \tilde{U}\Sigma\tilde{V}^T$ be two singular value decompositions of A ; that is,

$$\begin{cases} AA^T \mathbf{u}_i = \sigma_i^2 \mathbf{u}_i & , \quad AA^T \tilde{\mathbf{u}}_i = \sigma_i^2 \tilde{\mathbf{u}}_i & , \quad i = 1, \dots, m \\ A^T A \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i & , \quad A^T A \tilde{\mathbf{v}}_i = \sigma_i^2 \tilde{\mathbf{v}}_i & , \quad i = 1, \dots, n \end{cases}$$

and

$$\begin{cases} \mathbf{u}_i^T \mathbf{u}_j = \tilde{\mathbf{u}}_i^T \tilde{\mathbf{u}}_j = \delta_{ij} & , \quad i, j = 1, \dots, m \\ \mathbf{v}_i^T \mathbf{v}_j = \tilde{\mathbf{v}}_i^T \tilde{\mathbf{v}}_j = \delta_{ij} & , \quad i, j = 1, \dots, n, \end{cases}$$

where $U = [\mathbf{u}_1 \dots \mathbf{u}_m]$, $\tilde{U} = [\tilde{\mathbf{u}}_1 \dots \tilde{\mathbf{u}}_m]$, $V = [\mathbf{v}_1 \dots \mathbf{v}_n]$, and $\tilde{V} = [\tilde{\mathbf{v}}_1 \dots \tilde{\mathbf{v}}_n]$. Then

$$A^+ = V\Sigma^+U^T = \tilde{V}\Sigma^+\tilde{U}^T$$

where Σ^+ is defined in Eq. (61).

Proof. Write $\sigma_1, \dots, \sigma_k = \underbrace{\lambda_1, \dots, \lambda_1}_{i_1}, \dots, \underbrace{\lambda_\ell, \dots, \lambda_\ell}_{i_\ell}$, where $\lambda_1 > \dots > \lambda_\ell$ and $i_1 + \dots + i_\ell = k$. For each $j = 1, \dots, \ell$, let P_j be an $i_j \times i_j$ unitary matrix such that

$$[\tilde{\mathbf{u}}_{i_1+\dots+i_{j-1}+1} \dots \tilde{\mathbf{u}}_{i_1+\dots+i_j}] = [\mathbf{u}_{i_1+\dots+i_{j-1}+1} \dots \mathbf{u}_{i_1+\dots+i_j}]P_j$$

In addition, let $P_{\ell+1}$ be another $(n-k) \times (n-k)$ unitary matrix such that

$$[\tilde{\mathbf{u}}_{k+1} \dots \tilde{\mathbf{u}}_m] = [\mathbf{u}_{k+1} \dots \mathbf{u}_m]P_{\ell+1}.$$

Then we have

$$\tilde{U} = U \begin{bmatrix} P_1 & & \circ \\ & \ddots & \\ \circ & & P_{\ell+1} \end{bmatrix}.$$

Since by Eq. (53), we have:

$$\tilde{\mathbf{v}}_p = \frac{1}{\lambda_j} A^T \tilde{\mathbf{u}}_p \quad , \quad p = i_1 + \dots + i_{j-1} + 1, \dots, i_1 + \dots + i_j,$$

where $j = 1, \dots, \ell$, it then follows that

$$\begin{aligned} [\tilde{\mathbf{v}}_{i_1+\dots+i_{j-1}+1} \dots \tilde{\mathbf{v}}_{i_1+\dots+i_j}] &= \frac{1}{\lambda_j} A^T [\tilde{\mathbf{u}}_{i_1+\dots+i_{j-1}+1} \dots \tilde{\mathbf{u}}_{i_1+\dots+i_j}] \\ &= \frac{1}{\lambda_j} A^T [\mathbf{u}_{i_1+\dots+i_{j-1}+1} \dots \mathbf{u}_{i_1+\dots+i_j}] P_j \\ &= [\mathbf{v}_{i_1+\dots+i_{j-1}+1} \dots \mathbf{v}_{i_1+\dots+i_j}] P_j \end{aligned}$$

again by Eq. (53).

Let Q be an $(n-k) \times (n-k)$ unitary matrix such that

$$[\tilde{\mathbf{v}}_{k+1} \dots \tilde{\mathbf{v}}_n] = [\mathbf{v}_{k+1} \dots \mathbf{v}_n] Q.$$

Hence, we have

$$\tilde{V} = V \begin{bmatrix} P_1 & & & \textcircled{O} \\ & \ddots & & \\ & & P_\ell & \\ \textcircled{O} & & & Q \end{bmatrix}.$$

This yields

$$\begin{aligned} (\tilde{V}\Sigma^+\tilde{U}^T)^T &= \tilde{U}\Sigma^+\tilde{V}^T \\ &= U \begin{bmatrix} P_1 & & & \\ & \ddots & & \\ & & P_\ell & \\ & & & P_{\ell+1} \end{bmatrix} \begin{bmatrix} \frac{1}{\lambda_1} I_{i_1} & & & \\ & \ddots & & \\ & & \frac{1}{\lambda_\ell} I_{i_\ell} & \\ & & & \textcircled{O} \end{bmatrix} \begin{bmatrix} P_1^T & & & \\ & \ddots & & \\ & & P_\ell^T & \\ & & & Q^T \end{bmatrix} V^T \\ &= U \begin{bmatrix} \frac{1}{\lambda_1} I_{i_1} & & & \\ & \ddots & & \\ & & \frac{1}{\lambda_\ell} I_{i_\ell} & \\ & & & \textcircled{O} \end{bmatrix} V^T = U\Sigma^+V^T = (V\Sigma^+U^T)^T. \end{aligned}$$

or $\tilde{V}\Sigma^+\tilde{U}^T = V\Sigma^+U^T$ as required. Here, we have used the fact that $P_j P_j^T = I_{i_j}$. This completes the proof of the theorem.

Characterization of the Moore-Penrose Pseudoinverse

The Moore-Penrose pseudoinverse A^+ of an $m \times n$ matrix A can be computed by finding the eigenvalues of $A^T A$, the corresponding orthogonal eigenvectors $\{v_1, \dots, v_n\}$ and the null space of $A A^T$ since the other eigenvectors $\{u_1, \dots, u_k\}$ of $A A^T$ can be computed by using Eq. (51). In the following, we give a characterization of A^+ in terms of some matrix identities.

Theorem 2. A^+ is the Moore-Penrose pseudoinverse of A if and only if A^+ satisfies the following properties:

- (i) $AA^+A = A$,
- (ii) $(AA^+)^T = AA^+$,
- (iii) $A^+AA^+ = A^+$, and
- (iv) $(A^+A)^T = A^+A$.

Proof. It is clear from definition that A^+ satisfies the four properties listed. Now suppose that B satisfies (i) through (iv), and we have to prove that $B = A^+$. To do so, recall that

$$A = U \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} V^T$$

where

$$\Sigma_1 = \begin{bmatrix} \sigma_1 & & \bigcirc \\ & \ddots & \\ \bigcirc & & \sigma_k \end{bmatrix}.$$

We define

$$\begin{bmatrix} B_{9} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = V^T B U.$$

Then by using the property (i) for B , we have

$$ABA = A,$$

so that

$$(U^T A V)(V^T B U)(U^T A V) = U^T A V$$

or

$$\begin{bmatrix} \Sigma_1 & \bigcirc \\ \bigcirc & \bigcirc \end{bmatrix} \begin{bmatrix} B_9 & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} \Sigma_1 & \bigcirc \\ \bigcirc & \bigcirc \end{bmatrix} = \begin{bmatrix} \Sigma_1 & \bigcirc \\ \bigcirc & \bigcirc \end{bmatrix}$$

or

$$\begin{cases} \Sigma_1 B_9 \Sigma_1 = \Sigma_1 \\ \Sigma_1 B_{12} = 0 \end{cases}.$$

Therefore, $B_9 = \Sigma_1^{-1}$ and $B_{12} = 0$.

By using the property (ii) for B , namely: $(AB)^T = AB$ or $(U^T A B U)^T = U^T A B U$, so that

$$[(U^T A V)(V^T B U)]^T = (U^T A V)(V^T B U)$$

or

$$\left(\begin{bmatrix} \Sigma_1 & \bigcirc \\ \bigcirc & \bigcirc \end{bmatrix} \begin{bmatrix} \Sigma_1^{-1} & \bigcirc \\ B_{21} & B_{22} \end{bmatrix} \right)^T = \begin{bmatrix} \Sigma_1 & \bigcirc \\ \bigcirc & \bigcirc \end{bmatrix} \begin{bmatrix} \Sigma_1^{-1} & \bigcirc \\ B_{21} & B_{21} \end{bmatrix},$$

we have

$$\begin{bmatrix} I_k & \Sigma_1 B_{21}^T \\ \bigcirc & \bigcirc \end{bmatrix} = \begin{bmatrix} I_k & \bigcirc \\ B_{21} \Sigma_1 & \bigcirc \end{bmatrix}$$

so that $B_{21} = 0$. Similarly, by using property (iii) for B , we also have $B_{22} = 0$. Hence, we conclude that

$$B = V \begin{bmatrix} B_0 & B_{12} \\ B_{21} & B_{22} \end{bmatrix} U^T = V \begin{bmatrix} \Sigma_1^{-1} & \bigcirc \\ \bigcirc & \bigcirc \end{bmatrix} U^T = A^+.$$

Note that property (iv) has not been used since it is a consequence of (i)-(iii).

Application to Least-Squares Estimation

Let us now return to the linear system of Eq. (27) and using the notation defined in Eq. (46), we are lead to the system $A\mathbf{c} = \mathbf{b}$ where A is usually singular. Hence, the system may be inconsistent, at least numerically, and even if it is consistent, there are infinitely many solutions. As suggested in the beginning of this section, we will look for the (unique) least-squares solution with minimum norm; and by this, we mean the solution of the extremal problem in Eq. (48). The following result gives the solution.

Theorem 3. Let A^+ be the Moore-Penrose pseudoinverse of A and set $\mathbf{c}^* = A^+ \mathbf{b}$. Then

$$\|A\mathbf{c}^* - \mathbf{b}\|_{\ell^2} = \min_{\mathbf{c}} \|A\mathbf{c} - \mathbf{b}\|_{\ell^2}$$

$$\text{and } \|\mathbf{c}^*\|_{\ell^2} = \min\{\|\hat{\mathbf{c}}\|_{\ell^2} : \|A\hat{\mathbf{c}} - \mathbf{b}\|_{\ell^2} = \min_{\mathbf{c}} \|A\mathbf{c} - \mathbf{b}\|_{\ell^2}\}.$$

Proof. Let $A = U\Sigma V^T$ be a singular value decompositioin of A as described in the subsection entitled Singular Value Decomposition and the Moore-Penrose Pseudoinverse. Then

$$\begin{aligned} \|A\mathbf{c} - \mathbf{b}\|_{\ell^2} &= \|U\Sigma V^T \mathbf{c} - \mathbf{b}\|_{\ell^2} \\ &= \|\Sigma V^T \mathbf{c} - U^T \mathbf{b}\|_{\ell^2} \end{aligned}$$

since U is unitary, so that $\|U\mathbf{a}\|_{\ell^2} = \|\mathbf{a}\|_{\ell^2}$ for any vector \mathbf{a} . Write

$$V^T \mathbf{c} = (\gamma_1, \dots, \gamma_n)^T \quad \text{and} \quad U^T \mathbf{b} = (\beta_1, \dots, \beta_n)^T.$$

Then

$$\|\Sigma(V^T \mathbf{c}) - U^T \mathbf{b}\|_{\ell^2}^2 = (\sigma_1 \gamma_1 - \beta_1)^2 + \dots + (\sigma_k \gamma_k - \beta_k)^2 + \beta_{k+1}^2 + \dots + \beta_n^2.$$

Hence, the minimum of $\|A\mathbf{c} - \mathbf{b}\|_{\ell^2}$ is attained at $\tilde{\mathbf{c}} = (\tilde{\gamma}_1, \dots, \tilde{\gamma}_n)^T$ where $V^T \tilde{\mathbf{c}} = \left(\frac{1}{\sigma_1} \beta_1, \dots, \frac{1}{\sigma_k} \beta_k, \gamma_{k+1}, \dots, \gamma_n\right)^T$ with arbitrary $\gamma_{k+1}, \dots, \gamma_n$. In addition, since

$$\|\tilde{\mathbf{c}}\|_{\ell^2}^2 = \|V^T \tilde{\mathbf{c}}\|_{\ell^2}^2 = \left(\frac{1}{\sigma_1^2} \beta_1^2 + \cdots + \frac{1}{\sigma_k^2} \beta_k^2 \right) + (\gamma_{k+1}^2 + \cdots + \gamma_n^2),$$

the solution $\tilde{\mathbf{c}}$ with minimum ℓ^2 norm is attained when $\gamma_{k+1} = \cdots = \gamma_n = 0$, or at $\tilde{\mathbf{c}} = \mathbf{c}^*$, where

$$\begin{aligned} V^T c^* &= \left(\frac{1}{\sigma_1} \beta_1, \dots, \frac{1}{\sigma_k} \beta_k, 0, \dots, 0 \right) \\ &= \begin{bmatrix} \frac{1}{\sigma_1} & & & & & \\ & \ddots & & & & \\ & & \frac{1}{\sigma_k} & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} \end{aligned}$$

or equivalently,

$$c^* = V \Sigma^+ U^T b = A^+ b.$$

This completes the proof of the theorem.

Algorithms for determining singular value decompositions, and hence Moore-Penrose pseudoinverses A^+ , are available in the literature [6 - 12], for example.

ESTIMATION OF TIME-OF-ARRIVAL

As discussed in the subsection entitled Spline Representation of Underwater Acoustic Signals, the spline functions of degree k defined on the time interval $[0, d]$ with knot sequence

$$\mathbf{t}_h: t_0 < t_1 < \cdots < t_{n-1} < \cdots < t_{n+k},$$

where $0 \leq t_0 < d$, $t_n = d$, and $t_j = t_{j-1} + h$ for $j = 1, \dots, n+k$, with

$$h = \frac{d - t_0}{n}, \quad (62)$$

will be used to represent underwater acoustic signals. As in that subsection, the least-squares fit is used to determine the spline curve. Since a spline function is given by the spline series in Eq. (11), it vanishes identically on $[0, t_0]$ and "takes off" at t_0 (see Fig. 9). It is clear from this model that the initial knot t_0 represents the time-of-arrival. Hence, this knot must also be determined. While both n and d are fixed, with d denoting the length of the time interval and n the number of interior knots in the time interval, the relationship shown in Eq. (62) implies that determining t_0 is equivalent to determining h . Since the larger the value n is used the better the estimation becomes, it is advisable to choose a relatively large value of n , provided that the computational time is reasonable. Hence, in the mathematical model in Eq. (13), the minimization must be taken not only over the spline coefficients c_0, \dots, c_{n-1} , but also over the nonlinear parameter h . Furthermore, there are at least two reasons that we should choose the minimum value of those h that solve the optimization problem. First, if the "data function" happens to be piecewise linear with equally spaced knots, then the minimum h , or maximum t_0 , is the **exact** time-of arrival, while certain smaller estimates of t_0 still

reproduce the signal (see Example 1 in Appendix II). In addition, since h is the distance between two consecutive knots, the smaller the value of h , the better the approximation of $S_k = S_{k,h}$ [given by Eq. (11)] to the measured signal f is obtained. More precisely, we will study the following extremal problem. Let

$$K_k(h, \mathbf{c}) = \|f - \sum_{j=0}^{n-1} c_j B_{k,t_h,j}\|^2 + \lambda \left(\sum_{j=0}^{n-1} c_j^2 \right), \quad (63)$$

where $\mathbf{c} = (c_0, \dots, c_{n-1})$ and $\lambda > 0$ is fixed. Determine the set $H = \{\tilde{h}\}$ of values \tilde{h} such that

$$K_k(\tilde{h}, \tilde{\mathbf{c}}) = \inf_{h, \mathbf{c}} K_k(h, \mathbf{c}) \quad (64)$$

for some sequences $\tilde{\mathbf{c}} = (\tilde{c}_0, \dots, \tilde{c}_{n-1})$, where inf denotes the infimum (or “minimum”) and is taken over all possible $h > 0$ and all $\mathbf{c} = (c_0, \dots, c_{n-1})$. It should be emphasized that the minimization is taken independent of the order of h and \mathbf{c} . Let

$$h^* = h(k) = \inf H$$

be the greatest lower bound of the set $H = \{\tilde{h}\}$. Then by Eq. (62), the time-of-arrival of the acoustic signal with measurement f is given by

$$t_0 = d - nh^* \quad (65)$$

Existence, Uniqueness, and Characterization

As discussed in the subsection entitled Spline Representation of Underwater Acoustic Signals, both the L^2 and $\ell^2(\mathbf{w})$ norms will be used. Given any $\varepsilon > 0$, by the definition of infimum, there exists a pair (h_0, \mathbf{c}_0) where $h_0 > 0$, such that

$$K_k(h_0, \mathbf{c}_0) < \inf_{h, \mathbf{c}} K_k(h, \mathbf{c}) + \varepsilon. \quad (66)$$

Since

$$\begin{aligned} K_k(h_0, \mathbf{c}_0) &\geq \inf_{\mathbf{c}} K_k(h_0, \mathbf{c}) \\ &\geq \inf_{h>0} \{ \inf_{\mathbf{c}} K_k(h, \mathbf{c}) \}, \end{aligned}$$

we have

$$\inf_{h>0} \left\{ \inf_{\mathbf{c}} K_k(h, \mathbf{c}) \right\} < \inf_{h, \mathbf{c}} K_k(h, \mathbf{c}) + \varepsilon. \quad (67)$$

Since this inequality holds for any $\varepsilon > 0$, we may conclude that

$$\inf_{h>0} \left\{ \inf_{\mathbf{c}} K_k(h, \mathbf{c}) \right\} \leq \inf_{h, \mathbf{c}} K_k(h, \mathbf{c}). \quad (68)$$

However, since it is clear from definition that the quantity on the right-hand side is no greater than that on the left-hand side, it follows that the extremal problem we wanted to solve becomes an iterated extremal problem, namely:

$$\inf_{h,c} K_k(h, c) = \inf_{h>0} \left\{ \inf_c K_k(h, c) \right\}. \quad (69)$$

Now, for each fixed $h > 0$, since the extremal problem of finding a $\tilde{c}(h)$ such that

$$E_k(h) := K_k(h, \tilde{c}(h)) = \inf_c K_k(h, c) \quad (70)$$

is the (linear) least-squares problem discussed in the subsection entitled Spline Representation of Underwater Acoustic Signals, which, as we have seen, always has a solution by solving a system of linear equations, and in fact, this solution is always unique for L^2 , and also unique for $\ell^2(\mathbf{w})$ if the minimum norm (or Moore-Penrose pseudoinverse) solution is used, we see that the original extremal problem in Eq. (64) will also have a solution provided that there exists an $\tilde{h} > 0$ such that

$$E_k(\tilde{h}) = \inf_{h>0} E_k(h). \quad (71)$$

But the existence of \tilde{h} is clear, since $E_k(h)$ is a continuous function on $[0, d/n]$ and

$$E_k(0) = \|f\|^2$$

cannot be a minimum. Let H be the non-empty set of $\tilde{h} > 0$ that satisfies Eq. (71). The greatest lower bound of H , which is clearly a positive number denoted by h^* , is unique. Summarizing the above argument, we have the following.

Theorem 4. *There exist a unique $h^* > 0$ and a sequence $c^* = (c_0^*, \dots, c_{n-1}^*)$ such that*

$$K_k(h^*, c^*) = \inf_{h,c} K_k(h, c),$$

provided that the minimum-normed least-squares solution is used when the $\ell^2(\mathbf{w})$ norm is considered, where the quantity $K_k(h, c)$ is defined as in Eq. (63), and h^ is the greatest lower bound of all $\tilde{h} > 0$ that satisfies Eq. (64). Furthermore, (h^*, c^*) can be achieved as follows. For each $h > 0$, let $\tilde{c}(h)$ be the unique (minimum-normed) least-squares solution of Eq. (70). Find the set H of absolute minima of the continuous function $E_k(h)$ on $[0, d/n]$. (An absolute minimum must also be a relative minimum here.) Then*

$$h^* = \inf H,$$

and $c^ = c^*(h^*)$ is the (minimum-normed) least-squares solution of Eq. (70) for $h = h^*$.*

Recall that the time-of-arrival is given by $t_0 = d - nh^*$ in Eq. (65). The main difficulty in the procedure outlined above is solving the nonlinear problem of finding the absolute minima of $E_k(h)$; namely, the extremal problem in Eq. (71). For the continuous setting (or L^2 norm), this procedure will be greatly simplified by giving an explicit formulation of the error functional $E_k(h)$.

Estimation for the Continuous Setting

For any discrete data $(\tau_i, f_i), i = 1, \dots, N$ and $0 = \tau_1 < \dots < \tau_N = d$, let $\hat{f}(t)$ be the piecewise linear function on $[0, d]$, linear on each subinterval $[\tau_i, \tau_{i+1}]$, such that

$\hat{f}(\tau_i) = f_i$. Let \mathbf{f}_h^0 be the n -vector as defined in Eq. (21). By using the same change of variables as in the subsection entitled Computation of the Coefficient Matrices, we have

$$\mathbf{f}_h^0 = \begin{bmatrix} f_{h,0}^0 \\ \vdots \\ f_{h,n-1}^0 \end{bmatrix}, \quad f_{h,i}^0 = h \int_0^{n-i} \hat{f}(h(t-n+i)+d) N_k(t) dt,$$

for $i = 0, \dots, n-1$. Returning to Eqs. (20), (30) and (33), let $A_{k,h} = [b_{ij}^k]$, where

$$b_{ij}^k = h \int_0^{n-i} N_k(t) N_k(t+i-j) dt, \quad (72)$$

$i, j = 0, \dots, n-1$. Then $\tilde{\mathbf{c}} = \tilde{\mathbf{c}}(h)$ is the (unique) solution of the system of linear equations:

$$(A_{k,h} + \lambda I_n) \tilde{\mathbf{c}} = \mathbf{f}_h^0. \quad (73)$$

To study this system, set

$$\begin{cases} B = \left[\int_0^{n-i} N_k(t) N_k(t+i-j) dt \right]_{0 \leq i,j \leq n-1} \\ \mathbf{b} = \left[\int_0^{n-i} \hat{f}(h(t-n+i)+d) N_k(t) dt \right]_{0 \leq i \leq n-1}. \end{cases} \quad (74)$$

It is clear that B is a positive definite symmetric $n \times n$ matrix. Let

$$\Lambda = \begin{bmatrix} \lambda_1 & & \circlearrowleft \\ & \ddots & \\ \circlearrowleft & & \lambda_n \end{bmatrix} \quad (75)$$

be the diagonal matrix of eigenvectors of B where $\lambda_1 \geq \dots \geq \lambda_n > 0$ and form a unitary matrix P from the corresponding eigenvectors. Then $B = P \Lambda P^T$. Hence, Eq. (73) becomes:

$$(hP \Lambda P^T + \lambda I_n) \tilde{\mathbf{c}} = h\mathbf{b} \quad (76)$$

or

$$P(h \Lambda + \lambda I_n) P^T \tilde{\mathbf{c}} = h\mathbf{b}, \quad (77)$$

so that

$$\tilde{\mathbf{c}} = P \Gamma P^T \mathbf{b} \quad (78)$$

where

$$\Gamma = \begin{bmatrix} \frac{h}{h\lambda_1 + \lambda} & & \circlearrowleft \\ & \ddots & \\ \circlearrowleft & & \frac{h}{h\lambda_n + \lambda} \end{bmatrix}. \quad (79)$$

Note that both $\mathbf{b} = \mathbf{b}(h)$ and $\tilde{\mathbf{c}} = \tilde{\mathbf{c}}(h)$ are functions of h . The function $E_k(h)$ in Eq. (70) to be minimized now becomes:

$$E_k(h) = h\tilde{\mathbf{c}}^T B \tilde{\mathbf{c}} + \lambda \tilde{\mathbf{c}}^T \tilde{\mathbf{c}} - 2h\tilde{\mathbf{c}}^T \mathbf{b} + \int_0^d |\hat{f}(t)|^2 dt. \quad (80)$$

To simplify the computational procedure, it is recommended to set

$$\hat{\mathbf{b}} = P^T \mathbf{b} \quad (81)$$

where \mathbf{b} is defined in Eq. (74). Then Eq. (80) can be simplified to be

$$E_k(h) = \int_0^d |\hat{f}(t)|^2 dt - \sum_{i=1}^n \frac{h^2}{h\lambda_i + \lambda} \hat{b}_{i-1}^2 \quad (82)$$

where $\hat{\mathbf{b}} = (\hat{b}_0, \dots, \hat{b}_{n-1})$. It is now clear that $E_k(h)$ is differentiable with continuous derivative $E'_k(h)$ given by

$$E'_k(h) = - \sum_{i=1}^n \frac{h^2 \lambda_i + 2h\lambda}{(h\lambda_i + \lambda)^2} \hat{b}_{i-1}^2 - \sum_{i=1}^n \frac{2h^2}{h\lambda_i + \lambda} \hat{b}_{i-1} \hat{b}'_{i-1}. \quad (83)$$

It is also interesting to note that $E'_k(0) = 0$. Indeed, $h = 0$ gives a relative maximum $E_k(h)$. To determine the objective function $E_k(h)$ in Eq. (82), we must compute the eigenvalues $\{\lambda_1, \dots, \lambda_n\}$ of $B = B_n$ and the unitary matrix P formed by the corresponding (orthonormal) eigenvectors $\{u_1, \dots, u_n\}$, namely: $P = [u_1 \dots u_n]$. In the following, we give the example where $k = 1$.

Linear Spline Estimate

From Eq. (37), we see that for $k = 1$,

$$\begin{aligned} B = B_n &= \left[\int_0^{n-1} N_1(t) N_1(t+i-j) dt \right]_{0 \leq i, j \leq n-1} \\ &= \frac{1}{6} \begin{bmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ & & & 1 & 2 \end{bmatrix}. \end{aligned} \quad (84)$$

From Gershgorin's theorem, there is one eigenvalue in the interval [3.5], $n - 2$ eigenvalues in the interval [2,6], and one eigenvalue in the interval [1,3]. We can be much more precise by determining the eigenvalue and eigenvector pairs $(\lambda_i, \mathbf{u}_i)$, $i = 1, \dots, n$, more explicitly. To do so, let us consider the homogeneous linear system, where we have multiplied the matrix B by 6:

$$\begin{bmatrix} (4-\mu) & 1 & & & \\ 1 & (4-\mu) & 1 & & \\ & & (4-\mu) & 1 & \\ & & 1 & (2-\mu) & \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (85)$$

By setting $x = \frac{\mu}{2} - 2$, Eq. (85) can be reformulated as:

$$\left\{ \begin{array}{l} -2xy_1 + y_2 = 0 \\ y_1 - 2xy_2 + y_3 = 0 \\ \vdots \\ y_{n-2} - 2xy_{n-1} + y_n = 0 \\ y_{n-1} - 2(x+1)y_n = 0. \end{array} \right. \quad (86)$$

It is well known that the two linear independent solutions of the second order difference equations $y_{k+2} - 2xy_{k+1} + y_k = 0$ are the Chebyshev polynomials of the first and second kinds, $T_k(x)$ and $U_k(x)$, respectively, where

$$\left\{ \begin{array}{l} T_k(x) = \cos k \theta \\ U_k(x) = \frac{\sin(k+1)\theta}{\sin \theta} \end{array} \right. \quad (87)$$

with $\theta = \cos^{-1} x, 0 \leq \theta \leq \pi$. Since $U_0(x) = 1$ and $U_1(x) = 2x$ we have $y_k = U_{k-1}(x)$, $k = 1, \dots, n$, so that all except the last equation in Eq. (86) are satisfied. To satisfy the last equation in Eq. (86), we must have:

$$\frac{\sin(n-1)\theta}{\sin \theta} - (2x+1) \frac{\sin n \theta}{\sin \theta} = 0 \quad (88)$$

or equivalently,

$$\sin(n+1)\theta + 2 \sin n\theta = 0 \quad , \quad 0 < \theta \leq \pi. \quad (89)$$

This conclusion implies that $|x| \leq 1$ or $2 \leq \mu \leq 6$. However, as pointed out above, it follows from Gershgorin's theorem that there is an eigenvalue in Refs. 1 and 3 and hence, it is possible that $1 \leq \mu < 2$. To determine μ with $1 \leq \mu < 2$, θ may be considered to be complex. This is valid since $y_k = U_{k-1}(x)$ where $x = \cos \theta, \theta$ complex, still satisfies Eq. (86). A careful investigation reveals that a complex θ , which gives a real $x < -1$, is the only root other than the $(n-1)$ real roots of Eq. (89) in $(0, \pi)$.

Case 1. Real Solutions θ That Give the First $(n-1)$ Positive Eigenvalues:

Let

$$f(\theta) = \sin(n+1)\theta + 2 \sin n\theta. \quad (90)$$

Then we have :

(i)

$$\begin{aligned} f\left(j \frac{\pi}{n+1}\right) &= 2 \sin \frac{j n \pi}{n+1} \\ &= 2(-1)^{j+1} \sin \frac{j \pi}{n+1}. \end{aligned}$$

so that for $j = 1, \dots, n$,

$$\operatorname{sgn} f\left(j \frac{\pi}{n+1}\right) = (-1)^{j+1}. \quad (91)$$

(ii)

$$\begin{aligned} f\left(j \frac{\pi}{n}\right) &= \sin j\pi(1 + 1/n) \\ &= (-1)^j \sin \frac{j\pi}{n}, \end{aligned}$$

so that for $j = 1, \dots, n - 1$,

$$\operatorname{sgn} f\left(\frac{j\pi}{n}\right) = (-1)^j. \quad (92)$$

Now, since $j < n + 1$, we see that $j - 1/n < j/n + 1$; and therefore the intervals

$$\begin{aligned} I_1 &= \left(\frac{\pi}{n+1}, \frac{\pi}{n}\right), \\ I_2 &= \left(\frac{2\pi}{n+1}, \frac{2\pi}{n}\right), \\ &\vdots \\ I_{n-1} &= \left(\frac{(n-1)\pi}{n+1}, \frac{(n-1)\pi}{n}\right) \end{aligned} \quad (93)$$

do not overlap. Note also that these intervals all lie in $(0, \pi)$, and in view of Eqs. (91) and (92), there is one root θ_l of Eq. (89) in each $I_l, l = 1, \dots, n - 1$.

Case 2. Complex Solution θ That Gives the nth Positive Eigenvalue:

Let

$$g(t) = \sinh(n+1)t - 2 \sinh nt. \quad (94)$$

Observe that since there must be an eigenvalue μ in $(0, 2)$, or $x < -1$, we set $\theta = \pi + jt$ where $j = \sqrt{-1}$. This gives

$$\begin{aligned} f(\pi + jt) &= (-1)^{n+1} \sin(j(n+1)t) + 2(-1)^n \sin(jnt) \\ &= (-1)^{n+1} j \left[\sinh(n+1)t - 2 \sinh nt \right] \\ &= (-1)^{n+1} j g(t). \end{aligned}$$

That is, we must solve for the real t in

$$g(t) = \sinh(n+1)t - 2 \sinh nt = 0, 0 < t < \infty \quad (95)$$

and the (unique) root t_n [or complex root $\theta_n = \pi + jt_n$ of Eq. (89)] gives:

$$x_n := \cos(\pi + j\theta) = -\cosh t_n < -1 \quad (96)$$

or

$$\lambda_n := 4 + 2x_n < 2. \quad (97)$$

We can now make the following conclusion. Let $\theta_l \in I_l, l = 1, \dots, n - 1$, be solutions of Eq. (89) and t_n be the solution of Eq. (95). Then for $l = 1, \dots, n - 1$, recalling the factor $1/6$ in Eq. (84), the eigenvalue-eigenvector pairs of the matrix B are,

$$\begin{cases} \lambda_l = (4 + 2 \cos \theta_l)/6 \\ \mathbf{U}_l = \frac{1}{\left(\sum_{i=1}^n \sin^2 i\theta_l \right)^{1/2}} \begin{bmatrix} \sin \theta_l \\ \sin 2\theta_l \\ \vdots \\ \sin n\theta_l \end{bmatrix} \end{cases} \quad (98)$$

and for $l = n$, the eigenvalue-eigenvector pair is

$$\begin{cases} \lambda_n = (4 - 2 \cosh t_n)/6 \\ \mathbf{U}_n = \frac{1}{\left(\sum_{i=1}^n \sinh^2 it_n \right)^{1/2}} \begin{bmatrix} -\sinh t_n \\ \sinh 2t_n \\ \vdots \\ (-1)^n \sinh nt_n \end{bmatrix} \end{cases} \quad (99)$$

Hence, for $k = 1$ (the linear spline), the following algorithm can be used to estimate the time-of-arrival. In all the algorithms presented in this report, it should be remarked that the original extremal problem can be written as a nested extremal problem, as verified in the subsection entitled Existence, Uniqueness, and Characterization. That is, a linear regression is first performed by simple linear algebra to determine the spline coefficients, and then a nonlinear optimization procedure follows.

Algorithm I (Linear Spline Estimation of Time-of-Arrival Under Low Noise Condition).

- (1°) Compute $\lambda_1, \dots, \lambda_{n-1}, \lambda_n$ and $\mathbf{U}_1, \dots, \mathbf{U}_{n-1}, \mathbf{U}_n$ in Eqs. (98) and (99).
- (2°) Let $P = [\mathbf{U}_1 \dots \mathbf{U}_n]$ and compute $\hat{\mathbf{b}} = P^T \mathbf{b}$ where \mathbf{b} is given by Eq. (74).
- (3°) For exact data, use $\lambda = 0$, while the noisier the data, the larger positive λ is required. Fix a $\lambda \geq 0$ and determine the set of \tilde{h} such that
 $E_1(\tilde{h}) = \min_{h>0} E_1(h)$, where $E_1(h)$ is given by Eq. (82) with \hat{b}_{i-1} being the i^{th} entry of $\hat{\mathbf{b}}$ (which depends on h).
- (4°) Determine the smallest value $h^* > 0$ among all values \tilde{h} .

The time-of-arrival is given by $t_0 = d - nh^*$. [See Eq. (65)]. The choice of λ as a function of the noise of the signal will be studied in the forthcoming report [4].

In computing $\lambda_1, \dots, \lambda_{n-1}$, the "bi-section" method may be used to determine θ_i in I_i since the values of $f(\theta)$ have opposite signs at the two end-points of I_i . To compute λ_n , Newton's method may be used to search for the unique root $t_n > 0$ of Eq. (95).

In computing the data vector \mathbf{b} , and hence $\hat{\mathbf{b}}$ ($\hat{\mathbf{b}} = P^T \mathbf{b}$), note that it depends on h . In fact, since $k = 1$, the $(i+1)^{st}$ component of \mathbf{b} is given by

$$b_i = \int_0^1 t \hat{f}(h(t-n+i)+d) dt + \delta_i \int_1^2 (2-t) \hat{f}(h(t-n+i)+d) dt \quad (100)$$

for $i = 0, \dots, n-1$, where

$$\delta_i = \begin{cases} 1 & \text{for } i = 0, \dots, n-2 \\ 0 & \text{for } i = n-1. \end{cases}$$

The integrals in Eq. (100) can also be written as

$$\begin{aligned} b_i &= \frac{1}{h} \int_{d-(n-i)h}^{d-(n-i-1)h} \left(\frac{t-d}{h} + n - i \right) \hat{f}(t) dt \\ &\quad + \delta_i \frac{1}{h} \int_{d-(n-i-1)h}^{d-(n-i-2)h} \left(2 - \frac{t-d}{h} - n + i \right) \hat{f}(t) dt. \end{aligned} \quad (101)$$

Here, the factor $1/h$ which gives a factor of $1/h^2$ for \hat{b}_i^2 should not be evaluated since it cancels with h^2 in $E_1(h)$. [See Eq. (82)]. Recall that $\hat{f}(t)$ is the piecewise linear function determined by the interpolation condition $\hat{f}(\tau_j) = f_j$, where (τ_j, f_j) is the data set.

In determining \tilde{h} in (3°) , instead of minimizing the quantity $E_1(h)$, it is equivalent to maximizing the quantity

$$T(h) = \sum_{i=1}^n \frac{h^2}{h\lambda_i + \lambda} \hat{b}_{i-1}^2, \quad (102)$$

and as pointed out above, the numerator h^2 is cancelled out with the factor $1/h^2$ from \hat{b}_{i-1}^2 . Recall that \hat{b}_i depends on h . To facilitate the optimization process, the derivative of $T(h)$ may be used. Let $\mathbf{c} = [c_0 \dots c_{n-1}]^T$ and $\hat{\mathbf{c}} = [\hat{c}_0 \dots \hat{c}_{n-1}]^T = P^T \mathbf{c}$ where

$$\begin{aligned} c_i &:= \int_{d-(n-i)h}^{d-(n-i-1)h} \frac{d-t}{h^2} \hat{f}(t) dt \\ &\quad - \delta_i \int_{d-(n-i-1)h}^{d-(n-i-2)h} \frac{d-t}{h^2} \hat{f}(t) dt \\ &\quad - (1 - \delta_i)(n - i - 1) \hat{f}(d - (n - i - 1)h). \end{aligned} \quad (103)$$

Then we have

$$\begin{aligned} T'(h) &= \sum_{i=1}^n \frac{-\lambda_i}{(h\lambda_i + \lambda)^2} \left(h\hat{b}_{i-1} \right)^2 \\ &\quad + \sum_{i=1}^n \frac{2}{h\lambda_i + \lambda} \left(h\hat{b}_{i-1} \right) \hat{c}_{i-1}. \end{aligned} \quad (104)$$

Estimation with Splines of Arbitrary Degree

To estimate with spline functions of higher degree, the method derived in the subsection entitled Linear Spline Estimate cannot be applied; and hence, we must depend on numerical estimates of eigenvalue-eigenvector pairs directly. Let

$$B_n^k = \left[\int_0^{n-i} N_k(t) N_k(t + i - j) dt \right]_{0 \leq i, j \leq n-1} \quad (105)$$

where the integrals are computed using the procedure derived in the subsection entitled Computation of the Coefficient Matrices. For instance, when cubic splines are used we have

$$B_n^3 = \begin{bmatrix} C_{n-3} & D \\ D^T & E_3 \end{bmatrix} \quad (106)$$

where C_{n-3} , D , and E_3 are given in Eqs. (41), (42), and (43), respectively. Let

$$(\lambda_i^k, U_i^k), i = 1, \dots, n,$$

be the eigenvalue-eigenvector pairs of B_n^k where $\lambda_1^k \geq \lambda_2^k \geq \dots \geq \lambda_n^k > 0$ and each U_i^k is normalized to have unit length. The algorithm to determine the time-of-arrival by using a k^{th} degree spline curve can be described as follows.

Algorithm II (Estimation of Time-of-Arrival by Splines of Arbitrary Degree k under Low Noise Condition).

- (1°) Choose k (depending on the desirable smoothness) and compute the Bernstein coefficients of the B -spline N_k as in the subsection entitled Computation of B -Splines. (See Figs. 6 and 7).
- (2°) By using the formulae in Eq. (37) and following the procedure described in the subsection entitled Computation of the Coefficient Matrices compute the matrix $A_{k,h}$. [If a cubic spline curve is used, skip these two steps and use the formula in Fig. 10.]
- (3°) Compute the eigenvalue and eigenvector pairs $(\lambda_i^k, U_i^k), i = 1, \dots, n$, and $\lambda_1^k > \dots > \lambda_n^k > 0$, of the matrix $A_{k,h}$. (E.g. the routines in Ref. 13 may be used.)
- (4°) Let

$$P = [U_1^k \dots U_n^k]$$

and compute $\hat{\mathbf{b}} = P^T \mathbf{b}$ where \mathbf{b} is given by Eq. (74).

- (5°) For exact data, use $\lambda = 0$, while a positive λ may be used for noisy data. Fix a $\lambda \geq 0$, and determine the set of \tilde{h} such that

$$E_k(\tilde{h}) = \min_{h>0} E_k(h),$$

where $E_k(h)$ is given by Eq. (82) with \hat{b}_{i+1} being the i^{th} entry of \mathbf{b} (which depends on h).

- (6°) Determine the smallest value $h^* > 0$ among all values \tilde{h} .

Again a method to determine the value of λ as a function of the noise will be discussed in the forthcoming report [4]. For the time being, use $\lambda = .01$ or even a smaller value if the noise level is very low.

Estimation for the Discrete Setting

If the noise level is fairly high, algorithms I and II are not applicable since there is no reasonable criterion to determine the data function \hat{f} . In this case the ℓ^2 -norm with a relatively larger value of λ should be used. The value of $\lambda = 1$ is recommended, although it varies with the noise. See the forthcoming report [4] for a better choice of λ .

Let $(\tau_\ell, f_\ell), \ell = 1, \dots, N$, be a set of data information. Set

$$\tilde{A}_{k,h} = \begin{bmatrix} \tilde{b}_{ij}^k \end{bmatrix} \quad (107)$$

where \tilde{b}_{ij}^k is given by Eq. (31) with

$$B_{k,t_h,i}(t) = N_k \left(n - i - \frac{1}{h}(d - t) \right). \quad (108)$$

Define

$$\tilde{f}_h := \begin{bmatrix} \tilde{f}_0 \\ \vdots \\ \tilde{f}_{n-1} \end{bmatrix} := \begin{bmatrix} \sum_{\ell=1}^n f_\ell B_{k,t_h,0}(\tau_\ell) w_\ell \\ \vdots \\ \sum_{\ell=1}^n f_\ell B_{k,t_h,n-1}(\tau_\ell) w_\ell \end{bmatrix} \quad (109)$$

and let

$$A = \tilde{A}_{k,h} + \lambda I_n \quad (110)$$

as in Eq. (46). Then we have the following algorithm to determine the time-of-arrival by using a k^{th} degree spline curve and the $\ell^2(\mathbf{w})$ norm.

Algorithm III (Estimation of Time-of-Arrival with the ℓ^2 Norm for Noisy Data).

- (1°) Choose k (depending on the desirable smoothness) and compute the Bernstein coefficients of the B -spline N_k as in the subsection entitled Computation of B -Splines. (See Figs. 6 and 7).
- (2°) Compute the polynomial pieces of N_k by using the formula in Eq. (5) with a_{ij}^k being the Bernstein coefficients and $\varphi_{ij}^k(u, v)$ defined by Eq. (4), where u, v are given in Eq. (2) with $[a, b]$ being the corresponding interval of the polynomial piece. Set

$$B_{k,t_h,i}(t) = N_k \left(n - i - \frac{1}{h}(d - t) \right). \quad (111)$$

- (3°) Compute \tilde{b}_{ij}^k in Eq. (31) and \tilde{f}_i by using $w_\ell = 1, \ell = 1, \dots, N$, and $0 \leq i, j \leq n - 1$.
- (4°) Fix a positive value λ , say $\lambda = 1$ in Eq. (110). (The noisier the data, the larger the value of λ is recommended.) Determine an SVD (singular value decomposition) $A = U\Sigma V^T$ of A , where U and V are unitary and

$$\Sigma = \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_m & 0 \\ & & & \ddots \\ & & & 0 \end{bmatrix}. \quad (112)$$

where $\sigma_1 > \dots > \sigma_m > 0$.

(5°) Let \mathbf{u}_i be the i^{th} column of U and \mathbf{v}_i the i^{th} column of V . Compute $\mathbf{c}^* = [c_0^* \dots c_{n-1}^*]^T$ by using the formula

$$\mathbf{c}^* = \sum_{i=1}^m \frac{1}{\sigma_i} \mathbf{u}_i^T \tilde{\mathbf{f}}_h \mathbf{v}_i, \quad (113)$$

where $\tilde{\mathbf{f}}_h = [\tilde{f}_0 \dots \tilde{f}_{n-1}]$ has been computed in (3°).

(6°) Compute

$$K(h) = \sum_{i=1}^N \left(f_i - \sum_{j=0}^{n-1} c_j^* N_k \left(n - j - \frac{1}{h}(d - \tau_i) \right) \right)^2 + \lambda \sum_{j=0}^{n-1} (c_j^*)^2. \quad (114)$$

Here, we have used $w_i = 1$. The value of λ must be the same as the λ in (4°).

(A smaller or larger value of λ is used depending on the noise level of the data.)

(7°) Determine the set of \tilde{h} such that

$$K(\tilde{h}) = \min_{h>0} K(h). \quad (115)$$

(8°) Determine the smallest value $h^* > 0$ among all values \tilde{h} .

(9°) Compute $t_0 = d - nh^*$.

Then t_0 is the time-of-arrival. Note that algorithms from Refs. 6 to 9 can be used in step (4°).

ACKNOWLEDGEMENT: The author would like to thank David George for his many helpful comments and supply of several useful references.

REFERENCES

1. Barrow, D. L., C. K. Chui, P. W. Smith, and J. D. Ward, Unicity of best L_2 approximation by second-order splines with variable knots, Bull. Amer. Math. Soc. **83** (1977), 1049-1050.
2. de Boor, C., *A Practical Guide to Splines*, Springer-Verlag, New York, 1978.
3. Chui, C. K., *Multivariate Splines: Theory and Applications*, SIAM Publ., Philadelphia, 1988.
4. Chui, C. K., Estimation of time-of-arrival of underwater acoustic signals by spline functions II. Noisy data, Report to Naval Research Laboratory -USRD, Orlando, FL. To appear.
5. Chui, C. K. and M. J. Lai, Computation of box splines and B -splines on triangulations of nonuniform rectangular partitions, Approx. Th. and Appl., 1987.
6. Dongarra, J. J. et al., **Linpack User's Guide**, SIAM Publ., Philadelphia 1979 (Chapter 11).
7. Golub, G. H. and C. F. Van Loan, **Matrix Computations**, Johns Hopkins Univ. Press, 1985.
8. Golub, G. H. and C. Reinsch, Singular value decomposition and least squares solutions, Num. Math. **14**, 403-420, 1970.

NRL Memorandum Report 6572

9. Klema, V. C. and A. J. Laub, The singular value decomposition: its computation and some applications, IEEE Trans. Automatic Control AC-25, No. 2, April, 1980.
10. Nash, J. C., **Compact Numerical Methods for Computers**, Wiley, 1979, pp. 26-31.
11. Nash, J. C., A one-sided transformation method for the singular-value decomposition and algebraic eigenvalue problem, Computer J. 1975, vol. 18, pp. 74-76.
12. Press, W. H., B. P. Flannery, S. A. Teukolsky, W. T. Vettering, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, Cambridge, England, 1986.
13. Smith, B. T., J. M. Boyle, J. J. Dongarra, B. S. Gabow, Y. Ikebe, V. C. Klema, and C. B. Moler, **Matrix Eigensystem Routines - EISPACK Guide**, Springer-Verlag, Berlin, Heidelberg, New York, 1976.